UNCLASSIFIED

DTIC FILE COPY

SECUR

RT DOCUMENTATION PAGE

AD-A200 221

1a. RE UNCL	1b. RESTRICTIVE MARKINGS
2a. SE	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) AFOSR-TR- 88-0934
6a. NAME OF PERFORMING ORGANIZATION: California Institute of Technology; 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION AFOSR/NA
6c. ADDRESS (City, State, and ZIP Code) Pasadena, CA 91125	7b. ADDRESS (City, State, and ZIP Code) Building 410, Bolling AFB DC 20332-6448
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: AFOSR; 8b. OFFICE SYMBOL (If applicable): NA	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER AFOSR-87-0155
8c. ADDRESS (City, State, and ZIP Code) Building 410, Bolling AFB DC 20332-6448	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO.: 61102F; PROJECT NO.: 2307; TASK NO.: A1; WORK UNIT ACCESSION NO.:

DTIC ELECTE OCT 05 1988 S E

11. TITLE (Include Security Classification)

(U) INVESTIGATIONS OF THE MOTION OF DISCRETE-VELOCITY GASES BY CELLULAR AUTOMATA

12. PERSONAL AUTHOR(S)

Sturtevant, Bradford; Broadwell, James Eugene

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
FINAL	FROM ~~1-1-87~~ TO ~~31-8-88~~	2 Sep 88	41

16. SUPPLEMENTARY NOTATION

1 Jan 87 - 31 Dec 87

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	MOLECULAR GASDYNAMICS, DIRECT SIMULATION MONTE CARLO METHOD, CELLULAR AUTOMATA.
	20.04		

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A model of molecular gasdynamics with discrete components of molecular velocity has been implemented for parallel computation, and two test problems have been calculated. It is shown that fewer than ten values of each component of molecular velocity are necessary to produce accurate results in calculations by direct-simulation Monte-Carlo methods of rarefied-gas flows involving moderately strong shock waves. Thus significant savings in memory required to store the molecular velocities are realized. Most cellular automata intended to describe fluid motion simulate single-speed particles moving on square or hexagonal lattices. It is clear that with only one allowed molecular speed, temperature or energy cannot be specified independently of the velocity. The present paper describs the results of an exploratory investigation of heat conductiuon and shock wave formation with the two-dimentional model. The irreversible macroscopic behavior of this microscopically reversible system is also examined.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. J. Sakell	22b. TELEPHONE (Include Area Code): (202) 767-4936; 22c. OFFICE SYMBOL: AFOSR/NA

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

88 10 5 182

AFOSR-TR- 88-0934

Final Technical Report
Of Research Carried Out Under Contract Number AFOSR-87-0155

submitted to the

Office of Scientific Research, United States Air Force
Building 410
Bolling Air Force Base
Washington, D.C. 20332-6448

Investigations of the Motion of Discrete-Velocity Gases by Cellular Automata

1 January, 1987 - 31 December, ~~1988~~ 1987

Principal Investigator: B. Sturtevant
Co-Investigator: J. E. Broadwell

Graduate Aeronautical Laboratories
California Institute of Technology
Pasadena, CA 91125

September 2, 1988

Investigations of the Motion of Discrete-Velocity Gases

Abstract: 1) A model of molecular gasdynamics with discrete components of molecular velocity has been implemented for parallel computation, and two test problems have been calculated. When the molecular velocity components have integer values, and time is discretized for digital computation, the particles move on a regular array of points and the gas is called a 'lattice gas'. Calculations of molecular motions are thus simplified. The outcome of binary collisions between identical particles with discrete velocity components is determined by simple reflections about axes of symmetry in the center-of-mass system, so calculations of collisions are sped up. It is shown that fewer than ten values of each component of molecular velocity are necessary to produce accurate results in calculations by direct-simulation Monte-Carlo methods of rarefied-gas flows involving moderately strong shock waves. Thus significant savings in memory required to store the molecular velocities are realized.

2) Most cellular automata intended to describe fluid motion simulate single-speed particles moving on square or hexagonal lattices. In the latter case, two-dimensional low Mach number flows have been shown by Frisch *et al.* to obey the Navier-Stokes equations. These authors also discuss the various difficulties associated with the models, in particular, the restriction to low speeds. Furthermore, it is clear that with only one allowed molecular speed, temperature or energy cannot be specified independently of the velocity. d'Humiéres *et al.* describe what appears to be the simplest multi-speed model for flows in both two and three dimensions. The present paper described the results of an exploratory investigation of heat conduction and shock wave formation with the two-dimensional model. The irreversible macroscopic behavior of this microscopically reversible system is also examined.

1. Introduction

With the design of lifting vehicles powered by air-breathing engines, which take off at sea level and fly at hypersonic speed in the earth's upper atmosphere, the demands on methods for numerical simulation of flows over complex bodies have increased significantly. When conditions are such that the governing partial differential equations change type, mapping out the flow field around the vehicle requires that finite-difference solutions from different codes be overlaid, even in the simplest case of two-dimensional flow of a perfect gas. When complex chemical effects with widely differing relaxation times become important, new computational difficulties often arise. Under such circumstances simulation methods which are 'exact' at the molecular level become attractive, since, presumably, once the correct treatment of the physics and chemistry has been assured at the microscopic level, macroscopic fields will be calculated correctly even in cases for which the Navier-Stokes equations are invalid. The major disadvantage of using molecular methods to calculate continuum flow fields is their extreme inefficiency, due to the level of detail at which the calculations are made. Thus, molecular simulation methods are most often used to calculate flows where their application is necessary, *e.g.*, in rarefied gas flows. Nevertheless, for the reasons stated above, and also because it often happens that some parts of a flow field are rarefied while others are not, there has always been interest in extending the application of molecular simulation methods well into the continuum flow regime. The objective of the present work is to study methods for simplifying the molecular approach in order to make it more amenable to application to continuum flows. In particular, we examine the consequences of modeling flows with molecules that move with only a few, rather than a continuum of, different velocities. Such discrete-velocity models have in the past stimulated many fundamental studies in kinetic theory (see, for example Ref. 1), and recently their computer implementation as cellular automata (CA) has generated a great deal of interest[2]. The implementation in the present work of two discrete-velocity models, using methods for concurrent computation, has provided further insight into the physics of non-equilibrium gas flow.

In this work we are concerned with methods which *directly* simulate molecular motions, and in particular, with the Direct Simulation Monte-Carlo method[3] (DSMC), and with cellular automata. Such methods do not solve systems of partial differential equations, so the mesh can be independent of the coordinate system, and the calculation of flow over complex bodies is inherently simple. The present research is an investigation of simplifications of the DSMC model and of generalizations of the CA approach. In the former study, the emphasis is upon the influence of the simplification on the speed and accuracy of the calculation of supersonic flow fields. In the case of cellular automata, on the other hand, which in their present form constitute models of perhaps ultimate simplicity, our interest is in finding the minimum generalizations necessary for the treatment of compressible flow. The models treated in this work follow the simplifications of the Boltzmann equation introduced by Carleman[4] and Broadwell[5,6] in which the molecular velocity components are discretized. In the early work, the molecular velocities were prescribed *a priori* whereas, in our treatment of the DSMC (described in detail in the following section), the discretized velocities emerge as needed in the course of the computation. Our work on CA is described in § 3. We investigate a lattice gas in which *three* molecular speeds are allowed. This generalization of the conventional single-speed automaton[2] is such that the particles continue to move on a simple lattice but, now the concept of 'temperature' can be introduced.

2. Integer Direct-Simulation Monte-Carlo Method (IDSMC)

In work presented during a poster session at the 15th International Symposium on Rarefied Gasdynamics in 1986, we adapted the DSMC to massively parallel computation by writing a new code in the C programming language* and porting it to the Intel iPSC multicomputer system. The research version of the parallel DSMC which we presently use treats the molecules as elastic hard spheres. Phenomenological models of real-gas effects, such as vibrational relaxation and dissociation, which have been developed at other laboratories, could easily be incorporated into our codes. In all of our calculations we use an adaptive grid of cells which at every time step automatically remeshes itself in the vertical and horizontal directions to keep the average number of particles in the rows and columns of cells constant. By this means we aid load balancing among the processors of the computing machine, the most important consideration for the efficient use of parallel computation.

In the conventional DSMC a relatively small sample of molecules (typically tens of thousands to hundreds of thousands) is taken to represent a flowing gas. Space is divided into cells whose size $(\Delta x, \Delta y, \Delta z)$ is small compared to the mean free path λ, and time is discretized into steps Δt which are smaller than the mean molecular collision time τ. The calculation of molecular collisions is decoupled from the motion of the particles, and only the molecules within a given cell are considered as candidates for collisions. Only binary collisions are treated, so the gas is, by definition, dilute. During a given time step, collisions within a cell are calculated until the (known) collision frequency has been achieved. After collisions in all cells have been so calculated, the particles are moved in free flight to locations appropriate for the beginning of the next time step.

The goal of the modifications to the DSMC discussed here is to limit the amount of information about the molecular velocities that is developed, so as to speed up the calculations and free memory space for the treatment of more particles. The most direct way to insure that the method for limiting the information kept on velocities does not result in spurious generation or loss of momentum or energy is to carry out the simplification in a way that insures the conservation of momentum and energy in *every* collision, as in the conventional floating-point calculations. It is easy to do this for identical particles if the *components* of the molecular velocities are discretized. In the IDSMC the velocity components are integers. Though, in principle, the number of values which the integer velocity components can assume is infinite, in practice, the number depends on the integer size provided by the digital computer used for the numerical calculations. For 32-bit integers the number of velocities can be as large as 4×10^9, for 16 bits 65,536 and for eight bits 256. As will be seen below, the latter provides sufficient resolution for flows even up to hypersonic Mach numbers, so we usually declare the velocities to be one byte long.

---

* The C language is the most appropriate one to use in these applications because massively parallel computation is still in the early stages of development so that Fortran compilers generally have not been highly developed or, in some cases, are not available.

2.1. Discretization.

An immediate consequence of the discretization of the components of molecular velocity is that velocity is quantized with the unit of velocity, say q. In any given problem, whether q is 'small' or 'large' depends on the characteristic thermal speed of the molecules, say, the root mean squared thermal velocity,

$$c_s \equiv (\overline{c'^2})^{1/2} = \sqrt{3RT} \ , \tag{1}$$

where R is the gas constant and T is the local temperature. For high-temperature gases the velocity distribution function is 'wide', and many molecular speeds occur, so q is small compared to c_s. For cold gases the velocity distributior :nction is 'narrow', so only a few molecular speeds occur, and q may be of order c_s. In the IDSMC the number of molecular speeds found at any point in space and time adjusts to the temperature there. Furthermore, as in the DSMC, the number of speeds (the width of the distribution function) does not affect the computational cost. This is not the case when the gas is treated as a cellular automaton[2], where the computational cost increases rapidly with complexity.

Initially the velocity resolution is, in effect, set by the choice of the cell size, say Δx, compared to the distance traveled by a particle of speed q in time Δt, which we shall call the lattice spacing δ. For, as already stated, in the DSMC $\Delta t / \tau \equiv \mathrm{m}$ and $\Delta x / \lambda \equiv 1$ must both be somewhat smaller than unity. Furthermore,

$$\frac{\Delta x}{\delta} = \frac{\Delta x}{q\,\Delta t} = \frac{1}{\mathrm{m}}\frac{\lambda}{q\tau} = \frac{1}{\mathrm{m}}\frac{\overline{c'}}{q} \ , \tag{2}$$

where $\overline{c'} = \sqrt{8/3\pi}\, c_s$ is the mean thermal speed. Then for $1 \approx \mathrm{m}$, if δ is small compared to Δx, q must be small compared to $\overline{c'}$. Thus the velocity resolution improves as the cell contains more 'lattice sites'. In practice, Δx and Δt are chosen on the basis of λ and τ in the region of highest expected density and temperature in the flow under consideration*. In a typical example, we take

$$\delta = 0.1\lambda \ , \quad \Delta x = 0.5\lambda \ , \quad \Delta t = 0.2\tau \ ,$$

so $1 = 0.5$ and $\mathrm{m} = 0.2$, and, from (2), $\overline{c'} = 2q$, so

* When, as in the present work, the mesh is coarsened during the calculation in low density regions to keep 1 roughly constant, the right hand side of Eq. (2) increases because $1/\mathrm{m}$, not $\overline{c'}/q$, increases. Therefore, in this case the velocity resolution does not increase.

$$RT = \frac{\pi}{2} q^2 . \tag{3}$$

On the other hand, if we halve δ (to $= 0.05\,\lambda$) and increase Δt so that $l = m = 0.5$, then $\overline{c'} = 10\,q$ and the temperature is 25 times higher. In any application the dimensional value of q can be chosen to obtain the desired dimensional value of the reference temperature T, *e.g.*, 300K.

A further consequence of the discretization of the velocity components is that, if particles are initially distributed in space on a regular array co-incident with the axes of the co-ordinate system at points with spacing δ, then the particles remain on the array for all time, and we have a 'lattice gas'. By initially positioning the particles on a lattice, the spatial resolution is coarsened to a level consistent with the velocity resolution, and the calculation of particle motion is simplified and sped up; particle translations during the motion phase are obtained simply by counting lattice sites. Since the particle locations are integer numbers, storage requirements are also reduced.

In the IDSMC a (coarse) mesh of cells is superimposed on the (fine) lattice, and particles are drawn as candidates for collisions from all of the lattice sites within a cell. There may be as few as one lattice site in a cell, provided there are enough particles at the site that a sufficient number of collisions can be calculated during a time step to provide the necessary collision rate. This is in contrast to the procedure followed when the lattice gas is treated as a CA, in which case the number of particles at a site is limited by an exclusion principle, and every particle is treated as a candidate for a collision, consistent with a set of specified collision rules. For the CA the collision rate is an outcome of the calculation. In the IDSMC, as in the DSMC, a record is kept of the cell in which every particle resides, at the expense of additional storage. Since the adaptive grid of cells is superimposed on the fixed lattice, the cell sizes may not have any simple relationship to the lattice spacing, and two cells of the same size may contain a different number of lattice points. Thus it is necessary to take the cell volume to be proportional to the number of lattice sites in the cell in order that the time increment for each collision be properly computed. If there are many lattice sites per cell, the discretization of space is no longer significant. Thus, there is a continuum of IDSMC models of variable resolution.

2.2. *Collisions.*

To date we have considered only the interaction of identical hard-sphere molecules, and the following description of the method is limited to that case. As in the conventional DSMC, particle pairs are chosen for collision from among all the particles in a cell with a probability proportional to their relative velocity. The mechanics of collisions of integer-velocity molecules can be understood by the following considerations. For simplicity we present examples for a two-dimensional gas. The generalization to a three-dimensional gas is straightforward, and, except where otherwise noted, all of the calculations we present were performed with a three-dimensional gas. Figure 2.1 shows a simple quantitative example of a collision in which one of the colliding particles initially has velocity $(u_i, v_i) = (4q, q)$ and the other $(u_j, v_j) = (2q, 5q)$. The relative velocity is the vector difference between these two velocities, and the center-of-mass velocity lies at the center of the relative velocity vector. The consequence of momentum and energy conservation is that after the collision the relative velocity vector has the same magnitude and is simply rotated by the collision angle χ about the center-of-mass velocity. Candidates for outcome discrete velocities in the example of Figure 2.1 are indicated by small open circles on the perimeter of the large circle. It is clear that if one colliding particle is fast and the other is slow, as might occur in a high-temperature gas, the relative velocity will be large, so the diameter of the circle defined by the relative velocity vector will be large, and there will be many possible outcome velocities. Thus, in this method new integer velocities are generated or canceled automatically, as required by the local temperature and collision dynamics.

In general, four different cases must be considered depending on the parity of the components of the relative velocity vector. In Figure 2.1 both components are even (EE, *e.g.*, $(2q, 4q)$, with a squared relative velocity of $20q^2$). In this case the center-of-mass velocity falls on a lattice point in velocity space, and, in general, given a pair of initial velocities as shown, there are four possible pairs of output velocities (including that in which the initial velocity vectors are simply interchanged with no apparent change on the figure). The possible outcomes of the collisions are the intersections of the lattice in velocity space with the circle centered at the center-of-mass velocity of diameter equal to the relative velocity. It can be seen that the three pairs that are different from the input pair can be constructed by sequential reflections about the vertical, the 45° and the horizontal axes. We designate the slope of these three axes by 1/0, 1/1 and 0/1, respectively. If both components of the relative velocity are odd (OO, Figure 2.2), the center of mass falls at the center of a unit cell of the lattice in velocity space, and, again, four outcome pairs obtained by the same reflections as above are possible. On the other hand, when the relative-velocity components are of opposite parity (EO, Figure 2.3), the center of mass falls on the edge of a unit cell in velocity space, so no reflection about the 1/1 axis occurs, and only two outcome pairs are possible. In the EE and OO cases the relative velocity vector can lie at 45° (see Figure 2.4). In this case there are only two outcome pairs possible. On the other hand, in the EE and EO configurations the relative velocity vector can be vertical or horizontal (Figure 2.5). With EE there are only two outcome pairs, while with EO there is only one possible outcome, the same as the input.

At higher relative velocities, *i.e.*, at higher temperatures, reflections about other axes can be made, so more possibilities for outcomes arise. For example, the circle defined by a relative energy e_r of $130q^2$ intersects components $(7q, 9q)$ and $(3q, 11q)$, which cannot be obtained from each other by reflections about 1/0, 1/1/ or 0/1 (see Figure 2.6), so there are 8 possible outcome pairs for either of these OO input configurations. These outcomes can be constructed by reflection about axes of slope 1/2 and 2/1. In general, as the length of the relative velocity vector increases, symmetries about lines of slopes given by ratios of increasing values of whole numbers (*e.g.*, 1/3, 2/3, *etc.*) arise. In Figure 2.7 we indicate on the relative velocity lattice the number of points in the quadrant that are intersected by the circle about the origin which passes through that point. The boxed-in points are those that participate in symmetries more complex than 0/1, 1/0 and 1/1. If the relative velocity falls on an axis of symmetry, the possible outcomes are correspondingly reduced, and the case is degenerate. The algorithm for finding all the possible outcomes for a given relative velocity is relatively complex, so it is more efficient to do the calculation once and store the results in a look-up table for use during the Monte-Carlo calculations.

As the above discussion suggests, the number of possible collisions increases rapidly as the velocity resolution increases. In three dimensions there are $2^3 = 8$ parity cases, each possessing slightly different symmetry possibilities and degeneracies. Figure 2.8 is a plot of the number of integer points intersected by the spheres with squared radius from 1 to 50. Table 1 gives the first few entries of the look-up table used for the collisions.

Table 1. Sample from Look-up Table in Three Dimensions

Radius^2	Number of Points on Sphere	Coordinates of Points on Sphere
0	1	(0,0,0)
1	6	(-1,0,0) (0,-1,0) (0,0,-1) (0,0,1) (0,1,0) (1,0,0)
2	12	(-1,-1,0) (-1,0,-1) (-1,0,1) (-1,1,0) (0,-1,-1) (0,-1,1) (0,1,-1) (0,1,1) (1,-1,0) (1,0,-1) (1,0,1) (1,1,0)
3	8	(-1,-1,-1) (-1,-1,1) (-1,1,-1) (-1,1,1) (1,-1,-1) (1,-1,1) (1,1,-1) (1,1,1)
4	6	(-2,0,0) (0,-2,0) (0,0,-2) (0,0,2) (0,2,0) (2,0,0)
5	24	(-2,-1,0) (-2,0,-1) (-2,0,1) (-2,1,0) (-1,-2,0) (-1,0,-2) (-1,0,2) (-1,2,0) (0,-2,-1) (0,-2,1) (0,-1,-2) (0,-1,2) (0,1,-2) (0,1,2) (0,2,-1) (0,2,1) (1,-2,0) (1,0,-2) (1,0,2) (1,2,0) (2,-1,0) (2,0,-1) (2,0,1) (2,1,0)
6	24	(-2,-1,-1) (-2,-1,1) (-2,1,-1) (-2,1,1) (-1,-2,-1) (-1,-2,1) (-1,-1,-2) (-1,-1,2) (-1,1,-2) (-1,1,2) (-1,2,-1) (-1,2,1) (1,-2,-1) (1,-2,1) (1,-1,-2) (1,-1,2) (1,1,-2) (1,1,2) (1,2,-1) (1,2,1) (2,-1,-1) (2,-1,1) (2,1,-1) (2,1,1)
8	12	(-2,-2,0) (-2,0,-2) (-2,0,2) (-2,2,0) (0,-2,-2) (0,-2,2) (0,2,-2) (0,2,2) (2,-2,0) (2,0,-2) (2,0,2) (2,2,0)

This table contains the co-ordinates of the integer points on spheres centered at $(x, y, z) = (0, 0, 0)$ with differen. values of squared radius. Before using the table the velocities of the collision partners are transformed into the relative velocity frame. After choosing with uniform probability the outcome point from the table, under the condition that the parity of the components of the pre- and post-collision relative velocity vectors be the same, the final velocities are retransformed into the lab frame. Note that some spheres (*e.g.*, squared radius = 7) are not intersected at all by the lattice. The complete table presently used has entries for 1143 spheres. A look-up table containing entries for spheres with diameters up to, say, $40q$ can be several kilobytes in size. Since memory requirements of this magnitude are a matter of concern on many present-day parallel-computing machines in which the processors do not share memory, we choose to store the entries for only the octant $(x, y, z) > 0$ of each sphere, and reflect the selected outcome configuration of each collision across the planes $x = 0$, $y = 0$ and $z = 0$, each with 50% probability. This reduces the size of the look-up table to 1/8 of that needed for the full sphere, at the expense of a small increase in computational time.

It is clear from Figure 2.8 that, for collisions with moderately large relative velocities (typical of moderate temperatures), effectively a continuum of dynamical interactions is possible. On the other hand, for relatively low temperatures, where only a few velocities are used, the possible outcomes are limited, and it is easy to see that the resulting discrete distribution of, say, collision angle might be quite different from the expected uniform distribution of a continuous-velocity gas. Figure 2.9 is a histogram of the deflection angle of the relative velocity vector in an equilibrium three-dimensional IDSMC gas at two different temperatures. The figure was constructed by allowing 32,000 particles in a box to collide 64,000 times. The box contained just one computational cell, and the calculations were performed on a small sequential computer. The solid continuous line is a sine distribution appropriate for a hard-sphere gas. It can be seen that at the lower temperatures the angles 0, $\pi/2$, *etc.*, occur very often, and that the resulting distribution does not resemble that of a continuous-velocity gas, while at high temperatures the continuous distribution is well modeled. To a certain extent, the frequent occurrence of certain discrete collision angles compensates for the absence of neighboring values. However, in the integer-velocity gas there seem to be more occurrences of zero deflection angle than would account for the lack of small non-zero deflections. As will be seen below, at low temperatures (*i.e.*, low velocity resolution) the excess of zero-angle collisions has the effect of artificially increasing the diffusivity of the gas, while at moderate temperatures the effect on macroscopic quantities is not noticeable. The sampled distribution of particle velocity components at two different temperatures in an equilibrium gas shows no such anomalies, and agrees extremely well with the Maxwellian velocity distribution function (Figure 2.10). Even with the low-temperature case of Figure 2.10a, the nine available discrete-velocity components reproduce the Maxwellian distribution with excellent accuracy. We find, however, that the equilibrium distribution is not as sensitive a measure of the correctness of the collision process as the distribution function in a highly non-equilibrium flow.

It is worth noting that in cold regions of a flow the integer-velocity method imposes a minimum relative velocity between two low-energy particles. Thus, the amount that the time counter of the cell in which such a collision occurs can be incremented in cold regions is limited, while in a continuous-velocity gas very large time increments, effectively shutting down the cell for long periods, can occur. Note, also, that in cases for which there is only one site per cell there is no violation of the conservation of angular momentum if the lattice spacing is taken too large (see discussions by Meiburg[7] and Bird[8]).

2.3. Boundary conditions.

In the IDSMC, the boundaries, which in the calculations presented here are parallel to the lattice axes, are taken to lie midway between lattice points. Specular wall collisions are treated in the same way as in the conventional DSMC, by reflecting the particle trajectory across all wall segments necessary to insure that the particle at the end of the time step is inside the flow field, and by reversing its normal velocity after each reflection. For diffuse wall collisions the velocity components of the emitted particles are chosen from the integer Maxwellian distribution (*cf.* Fig. 10) corresponding to the wall temperature. The trajectories of colliding particles approaching and departing from walls are calculated with floating-point precision, and at the end of the time step the particle position is rounded to the nearest lattice site.

2.4. Results - IDSMC.

In this section we present the results of calculations of two test problems which exhibit certain features of the IDSMC method.

2.4.1. Relaxation to Equilibrium. In this calculation, 16,000 particles in a box consisting of one computational cell are initially distributed bimodally with integer-valued velocity components, as indicated in the top rows of Figures 2.11 and 2.12. The x-component molecular velocities are distributed in two narrow bands (each with RT $\approx 1.5q^2$) about $u = \pm 10q$, while the y- and z-component velocities have only one peak of the same width. The fluid is uniform in these calculations, and all particles in the box are candidates for collisions. Only the collisions are calculated; the particle motions are not. Figure 2.11 shows the results for the conventional DSMC method, while Figure 2.12 gives the results for the IDSMC at the same times. Though in the DSMC calculation the initial distribution contains only integer-valued components, after the first collision the velocities become decimal numbers. In Figure 2.11 the distributions are plotted as histograms with bin size q, while in Figure 2.12 the plotted spikes represent the accumulated data at the corresponding discrete values of velocity. The molecular velocities are sampled after 1 (second row), 2 (third row), and 10 collisions per particle (fourth row), respectively. It can be seen that in both calculations the initial bimodal distribution evolves into Maxwellian distributions with temperature $RT = 35.02\, q^2$ in all three directions (indicated by the solid curves in the bottom row), and that the IDSMC is essentially the same as the DSMC result.

Using the Bhatanagar-Gross-Krook model of a discrete-velocity gas, Broadwell[9] showed that the equilibrium distribution has the form of a Maxwellian, *i.e.*, for a two-dimensional stationary gas,

$$n_i \propto \exp\left[-A\left[u_i^2 + v_i^2\right]\right], \tag{4}$$

where the subscript $()_i$ refers to the class of particles with discrete velocity components u_i and v_i. In the limit in which all velocities are allowed, A becomes $(2RT)^{-1}$. Thus the distributions shown in the bottom row of Figure 2.12 are expected. The present calculations show that in addition the discrete and continuous distributions are very similar in non-equilibrium flows.

2.4.2. The normal shock wave. One way to rigorously test the new discrete-velocity method is to compare results for the structure of strong shock waves with DSMC calculations carried out on the same machine using comparable code with the same time step size, cell size, *etc.*. Figure 2.13 shows the normaiized density and temperature profiles obtained by the DSMC (solid line) and the IDSMC (points) for a normal shock wave of strength $M_s = 6.11$ in a perfect hard-sphere gas. The space co-ordinate is normalized with the upstream mean free path. Also shown are discrete Maxwellian distributions of molecular thermal velocities corresponding to the measured uniform states upstream and downstream of the shock together with the continuous Maxwellian, for comparison. The calculation is carried out in a nonsteady frame; the left wall of a box is impulsively accelerated to a constant speed of $8q$ at time $t = 0$, and the shock profile is sampled after 180 time steps of size $\Delta t = 0.1085\,\tau_1$, where τ_1 is the upstream mean free time. The cell size is about $\Delta x = 0.5\,\lambda$. There are about 60 particles in each cell and 20 lattice sites per mean free path. In order to achieve a smooth profile using this approach, a total of about 140 million collisions between 6.1 million particles are calculated, using 64 processors of an Intel iPSC message-passing multicomputer. Physical space is assigned to the computer nodes in accordance with the load-balancing algorithm already described. When, during the move phase of the calculation, a particle moves from one node's domain to that of another, it is sent there as a message. It can be seen that excellent agreement with the continuous-velocity DSMC model is achieved starting with only 7 values of each velocity component. In the uniform gas behind the shock 29 values of each component are found.

Figure 2.14 shows results from a similar problem, but with the upstream temperature in the IDSMC calculation 4 times smaller than in Figure 2.13. In this case the velocity resolution is poor, so the discrete calculation does not agree as well with the continuous-velocity result. As discussed above, with just a few possible velocities, zero deflection angle occurs too often in the discrete calculations, so the particle diffusivity is too large. Thus, the hot downstream particles diffuse toward the front of the shock, and the shock becomes too thick.

A sensitive test of the absolute accuracy of shock structure calculations is obtained by plotting the normal component of the pressure tensor, $p_{xx} = \rho \overline{u'^2}$, versus the specific volume. According to the x-momentum equation, this should be a straight Rayleigh line. In Figure 2.15 p_{xx}, normalized by its upstream value, which by definition is the upstream pressure, is plotted versus ρ_1/ρ for the same shock calculations as presented in Figure 2.13. The cluster of points at (1, 1) are from samples near the upstream end of the shock and the cluster near (46.4, 0.27) are from the downstream end. In the calculations reported here the cell size and time step were selected to achieve the performance indicated in the figure; larger values would have resulted in an S-shaped curve which deviated more from the straight Rayleigh line. The figure shows that, with the same time step and cell size, the IDSMC and the DSMC perform about the same.

2.5. Summary and Discussion, - IDSMC.

It has been shown that, for rarefied gasdynamics problems in which there are fairly strong shock waves, accurate results in calculations by direct-simulation Monte-Carlo methods can be obtained when the upstream distributions of each component of molecular velocity contain fewer than ten discrete values. When the molecular velocity components have integer values, the particles move on a lattice, so calculations of their motions are simplified. The outcome of binary collisions between identical particles with discrete velocity components is determined by simple reflections about axes of symmetry in the center-of-mass system, and the results can be stored in look-up tables for rapid use during calculations of collisions.

Calculations of relaxation to equilibrium and of shock wave structure have been performed to test the accuracy of the discrete-velocity method. Comparison has been made with known equilibrium results, and, for nonequilibrium, with calculations under identical conditions by the conventional DSMC. A test of absolute accuracy by plotting the Rayleigh line in normal shocks has been introduced. From these tests the velocity resolution cited above, necessary for achieving high accuracy, was deduced.

In the multicomputer used for the present calculations, provision for 256 different values of velocity, by declaring the molecular velocity components to be one byte long, allows 2.4 times more particles to be treated in a two-dimensional flow than when the particle velocities are stored as real numbers. With real numbers the storage per particle is 22 bytes (12 for the 3 components of velocity, 8 for 2 positions, and 2 for the particle index), while with the present method it is 9 bytes (3, 4 and 2). Thus flows with Reynolds numbers 2.4× greater can be calculated. Clearly, since many fewer than 256 velocity values are necessary, this result could be improved if the velocities were stored more compactly. For example, if only 5 bits were used for velocities and 10 bits for positions, the improvement could, in principle, be 3.5×.

By design, the IDSMC computes primarily with integer arithmetic and the DSMC with floating-point arithmetic. Therefore, comparison of the relative *speed* of the two methods is bound to be machine dependent. For example, in the calculation of relaxation to equilibrium reported above, in which only collisions in one large cell are calculated, and the particles are not moved, the IDSMC ran about 3× faster on a sequential SUN 3/60 microcomputer, which is apparently relatively efficient at integer calculations, than did the DSMC. On the other hand, in the strong shock wave problem on the Intel iPSC multicomputer, which is evidently a rather efficient floating-point machine, the IDSMC ran only about 10% faster. In view of the fact that our codes have not yet been optimized for speed, it is clear that more work needs to be done to define definitive benchmarks.

3. Study of a Multi-Speed Cellular Automaton

3.1. Cellular Automata.

The macroscopic behavior of a fluid near equilibrium is expected to be nearly independent of the details of the motion of the molecules which constitute it. For example, low Mach number flow of a gas and of a liquid are described by the same equations. This idea forms the basis for the cellular automaton (CA) simulation of fluids[2]. The aim of this approach is to maximally simplify the molecular dynamics while retaining the essential physics. This simplification of the molecular dynamics involves a full discretization of phase space *i.e.*, of both velocities and positions. This is in contrast with the discrete-velocity models[1,4,5,6], where only the velocity space is discretized, and the various finite difference equations where only position space is discretized. Discretization of both velocity and position gives rise to the notion of discrete time, the unit of time being that taken by the slowest moving particle to travel the smallest unit of distance in the direction of its velocity. All other particles move an integer number of link lengths in the direction of their velocities in the same time. The evolution of the system is then reduced to a set of discrete move and collide phases. At each instant of time, each lattice site collects the relevant information from its nearest neighbors (*i.e.*, the particles convect) and performs a simple transformation on it (*i.e.*, the particles collide). Such a limiting simplification is a cellular automaton and is implemented by mapping onto a digital computer.

To study the simplest of these models, only a few velocities are considered. Particles are then identified by their velocities, so that we have a small number of distinguishable particle types. For no reason other than to keep computation per time step small, an exclusion principle is adopted, namely, no lattice site is allowed to have more than one particle of a particular type. In the present work the rules for collisions conserve mass, momentum and energy. The choice of candidates for collision is arbitrary and thus the collision rate of CA, and therefore the mean free path, is model dependent. This is in contrast to the procedure used in Monte-Carlo methods used for directly simulating molecular flows[3], in which candidates are chosen to insure that the collision rate is correct.

3.1.1. Implementation. The implementation of cellular automata on a digital computer is simple, elegant and highly efficient. In the present work a lattice site is represented by a computer word. The computational domain is then an array of words. A particle of a particular type (*i.e.*, a particle with a certain velocity) is identified with a particular bit in the word. A word therefore has to have at least as many bits as there are velocities in the model. The presence or absence of a type of particle at a lattice site is indicated by the presence or absence (on or off) of the corresponding bit in the word representing the lattice site. When only a few velocities are present in the model, the move phase is accomplished by a small number of simple binary operations on the array of words representing the computational domain, while the collide phase is reduced to a table look-up. With more velocities, the move phase requires more binary operations, and the look-up table becomes bulky, necessitating a functional implementation of the collisions. In a variant of the above implementation, the presence or absence of a particle of a particular type at a group of lattice sites is compacted into a word. A set of words, as many as the number of different velocities, then represent several lattice sites, as many as the size of the word. In this scheme, the move phase amounts to shifting words bitwise in the appropriate direction and the collision phase to the evaluation of Boolean functions representing the collisions.

In either case, the simplicity of the move and collide steps makes it possible to simulate huge numbers of particles, in comparison to other direct simulation methods. Further, since only nearest neighbors interact, the evolution itself is highly localized and hence is ideally suited for concurrent computation. The communication overhead between the nodes of a parallel processor is proportional to the ratio of the perimeter of the physical space represented by a node to its area, *i.e.*, to the inverse of the aspect ratio of the computational domain. The complete synchrony between the various parts of the computational domain obviate the need for balancing load between the various processors dynamically.

The present study has concentrated on the simulation of two-dimensional fluids because most essential ideas can be described with them. An extension to three dimensions is straightforward.

3.2. Single- and Multiple-Speed Models.

In the classical CA, known as the HPP model[10], and in most subsequent models, the particles move with a single speed. Figure 3.1 shows the lattice corresponding to the HPP model, in which the particles can move in 4 different directions. The particles move along the horizontal or vertical links to cover one link length in one unit of time. The only non-trivial collisions are the horizontal and vertical head-on collisions. Then, if all particles are initially on the lattice, they remain so. Figure 3.2 shows a second single-speed model with a better symmetry, the FHP model[2]. Particles now have one of 6 different velocities, all of the same speed. There are now a number of non-trivial collisions but spurious conservations by 2-body collisions alone necessitate the implementation of three body collisions as well. It has been shown by Frisch *et al.*[2,11] two dimensions and d'Humieres *et al.*[12] for three dimensions that for the single-speed model on the hexagonal lattice of Figure 3.2 the fluid mechanical velocities, obtained by averaging over many sites, satisfy the incompressible Navier-Stokes equations. (On the square lattice (Figure 3.1), the cross-derivative convective terms in the averaged momentum equation are missing, leading to a spurious conservation of momentum[2].) The problem, however, with single-speed models is that the temperature cannot be represented independently of the velocity. In a two-speed model, even though differing proportions of the particles with the two allowable speeds can represent differing amounts of energy, there is no mechanism by which, in collisions conserving momentum and energy, particles can change speed. This implies that there is no dynamic balancing between the particles of differing speeds or that equilibration is only partial. Thus the simplest model with temperature as an independent degree of freedom is a three-speed model. Preliminary investigation of one such model, a 3-speed, 9-velocity model is presented here.

A major shortcoming of these lattice gases in modeling real fluids is that the state variables of such a gas depend on the frame of reference, so they are not Galilean invariant. Addition of more velocities will of course mitigate this problem. Even with just a few velocities, however, at very low speeds compared to the particle speeds (*i.e.*, in the low Mach number), the effect of Galilean non-invariance is negligibly small.

3.2.1. The 9-Velocity Model. Figure 3.3 shows the allowable velocities in the three-speed model, and Figure 3.4 the two dimensional lattice on which the particles move. The slow particles, which have unit speed, say q, are restricted to move on the horizontal and vertical links, while the fast particles, which have a speed of $\sqrt{2}\,q$, move on the diagonal links. The zero speed particles exist only to take part in collisions, to allow interaction between the other two speeds. Each lattice site has 8 nearest neighbors, 4 at a distance δ away and 4 others a distance $\sqrt{2}\,\delta$ away where δ is the distance traveled at speed q in unit time, Δt. To model a dilute gas with short-range intermolecular interactions, collision rules are defined in which only the nearest neighbors influence a lattice site. Then all possible collisions, each conserving mass, momentum and energy, of the types shown in Figure 3.5, take place, subject to the exclusion condition. Since only one particle of a given velocity is allowable at a site, the site may not be able to accommodate some of the particles resulting from some collisions and hence those collisions are excluded.

As stated above, the transport coefficients of a lattice gas depend on the details of the microscopic dynamics. To examine this behavior the experimental simulations have been carried out with two different set of rules (collisions). In the first (rule set 1), only binary collisions are implemented *i.e.*, a collision occurs at a site if and only if there are exactly two particles at that site and if at least one component of their momenta are oppositely directed. This set of rules therefore implements just 10 direct collisions and their inverses, so a total of 20 states of the $2^9 = 512$ possible at any lattice site are changeable. In the second (rule set 2), a collision occurs at a site, irrespective of the number of particles present there, providing only one collision is possible. Thus in some of these collisions two particles collide while others at the same site are not affected. The same two-body collisions are implemented in this rule set as in rule set 1, but this rule set transforms a total of 156 of the possible 512 states at a lattice site to a different configuration. Thus the collision rate is substantially increased. Other rules are possible, including that implementing all possible conservative collisions at a site in a certain specific order, or randomly, but the two rule sets described above are sufficient for our arguments.

3.2.2. The Boltzmann equations. An approximate description of the evolution of spatial averages of the nine populations in the automaton is given by Boltzmann equations for the corresponding discrete velocity gas. They are formulated by assuming that (1) the gas is dilute, so that only binary collisions are important, (2) there is no exclusion, and (3) molecular chaos prevails, *i.e.*, that the joint two-particle distribution function can be replaced by the product of two one-particle distribution functions. The Boltzmann equations for the 9-velocity model are given schematically in Table 1.

Table 1
Boltzmann Equations
9 Velocity Model

c_r =	$\sqrt{2}q$				$2q$	$2\sqrt{2}q$	$\sqrt{5}q$			
	$n_1n_3-n_0n_2$	$n_3n_5-n_0n_4$	$n_5n_7-n_0n_6$	$n_7n_1-n_0n_8$	$n_3n_7-n_1n_5$	$n_4n_8-n_2n_6$	$n_5n_2-n_1n_4$	$n_5n_8-n_1n_6$	$n_3n_8-n_7n_2$	$n_5n_6-n_7n_4$
$\frac{\partial n_0}{\partial t}$ =	+	+	+	+						
$\frac{\partial n_2}{\partial t}+q\frac{\partial n_2}{\partial x}+q\frac{\partial n_2}{\partial y}$ =	+					+	−		+	
$\frac{\partial n_4}{\partial t}-q\frac{\partial n_4}{\partial x}+q\frac{\partial n_4}{\partial y}$ =		+				−	+			+
$\frac{\partial n_6}{\partial t}-q\frac{\partial n_6}{\partial x}-q\frac{\partial n_6}{\partial y}$ =			+			+		+		−
$\frac{\partial n_8}{\partial t}+q\frac{\partial n_8}{\partial x}-q\frac{\partial n_8}{\partial y}$ =				+		−		−	−	
$\frac{\partial n_1}{\partial t}+q\frac{\partial n_1}{\partial x}$ =	−			−	+		+	+		
$\frac{\partial n_5}{\partial t}-q\frac{\partial n_5}{\partial x}$ =		−	−		+		−	−		
$\frac{\partial n_3}{\partial t}+q\frac{\partial n_3}{\partial y}$ =	−	−			−				−	−
$\frac{\partial n_7}{\partial t}-q\frac{\partial n_7}{\partial y}$ =			−	−	−				+	+

The gain and loss terms which appear on the right-hand sides of the equations for the 10 different collisions are indicated in the second row, while the relative velocity c_r for each of the collisions is indicated in the first row. The collision rate for each collision is proportional to c_r. The left hand side of each of the nine equations for the nine classes of particles are indicated in the first column. The entries in the matrix give the sign with which the gain/loss terms appear in the corresponding equation.

3.3. Observations and Discussion.

3.3.1. The equilibrium state. A first step towards understanding the nature of the lattice gas is to study its equilibrium. If there are no spatial gradients, these equations have solutions that approach a steady equilibrium state. This behavior is conveniently described by defining the H function

$$H = \sum_i n_i \log n_i \quad , \tag{1}$$

and showing that it must decline to a constant. According to the H Theorem, at thermodynamic equilibrium detailed balancing prevails, so the gain/loss combination from each collision goes to zero individually. These are the equilibrium equations for the model. They make up the following set of 5 independent equations.

$$\begin{aligned} n_1 n_3 &= n_0 n_2 \\ n_3 n_5 &= n_0 n_6 \\ n_5 n_7 &= n_0 n_6 \\ n_7 n_1 &= n_0 n_8 \\ n_3 n_7 &= n_1 n_5 \end{aligned} \tag{2}$$

Customarily, equilibrium conditions are expressed in terms of thermodynamic variables of state, in this case the density n, the energy of the system e_t and the mean velocities $\bar{u}$ and $\bar{v}$. Then, the definitions of these variables provide four other equations to be satisfied,

$$n = n_0 + n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8$$

$$\bar{u} = \frac{\left(n_1 + n_2 + n_8 - n_4 - n_5 - n_6\right)}{n} \tag{3}$$

$$\bar{v} = \frac{\left(n_2 + n_3 + n_4 - n_6 - n_7 - n_8\right)}{n}$$

$$e_t = \frac{q^2}{2n}\left(n_1 + n_3 + n_5 + n_7 + 2\left(n_2 + n_4 + n_6 + n_8\right)\right) \quad .$$

Thus there are nine equations for the nine variables n_i, and the system is uniquely determined.

The equilibrium equations are solved for some simple cases in Appendix I. Figure 3.6 shows the evolution of the H function for lattice-gas particles in a box initially distributed randomly with the state variables set to the desired final state. It is seen that, though the initial condition is in the proper macroscopic state, the system is initially out of equilibrium, but equilibrates in a few collisions. Experimental simulations of the equilibrium conditions are in accord with the solution of the above set of equations, as shown, for example in Figure 3.7.

3.3.2. Characterization of the Mean Free Path. The mean free path is of the order of the lattice spacing in the range of densities of interest. Above a certain density, the effects of exclusion come in strongly and the mean free path between effective collisions becomes very large. The mean free path is calculated in the following way. The initial condition in a box with doubly periodic boundary condition is set to correspond to prescribed values of density, temperature and velocity The boundary conditions were made doubly periodic because otherwise, shocks and rarefactions, in the cases of non-zero macroscopic velocity, dissipate directed kinetic energy and change the thermodynamic state of the system. The initial particle distributions are calculated numerically under the simplifying assumptions of binary collisions and absence of exclusion. In the simulation, exclusion may cause the actual particle distribution to be slightly different from that calculated initially. After the system is allowed to relax to equilibrium, the number of collisions in the box is counted for a number of time steps. Also kept track of is the ratio of particle population of different speeds. Then under the assumption of ergodicity of the behavior of particles, the typical collision frequency ν, and the mean particle speed $\bar{c}$ are calculated and the mean free path obtained from the relation $\lambda = \bar{c}/\nu$.

The variation of the mean free path with density, temperature and velocity for the two different rule sets was studied. As can be seen in Figure 3.8 and 9, the behavior of the mean free path with the two rule sets differs appreciably only in the regions of high density. Recall that the two rule sets implement the same collisions. The only difference between them is in the number of collisions which are effective.

A power regression between the mean free path and the density at low densities gives $\lambda \propto 1/\rho$ as in real gases (see Figures 3.8 and 3.9). The anomalous behavior at higher densities comes from the exclusion of some collisions. This study gives bounds on the density that can be used in these cellular automatons to study gas dynamics. The variation of the mean free path with temperature comes from the collision rules and probably cannot be given physical significance. Figure 3.10 illustrates the Galilean non-invariance of the model, *i.e.*, the mean free path changes with velocity. Note, however, that changes are small up to velocities of about $0.3q$.

3.3.3. Heat Conduction. To examine the nature of the model and, in particular, its ability to deal with heat conduction, we have studied heat transfer between two walls at rest (Figure 3.11). The walls are maintained at different temperatures and the temperature distribution in the gas is computed as a function of time. Temperatures are maintained by imposing, at each instant of time at the row of wall lattice sites, the equilibrium particle distribution corresponding to the wall conditions. The parameter of interest in the problem is the Knudsen number, $Kn = \lambda/L$, where L is the wall separation. The main aim is to determine the nature of the steady state profiles and the effect of a variation of Kn on these profiles. Both rule sets were used for some runs. The lt : cold wall was maintained at temperature of $0.6q^2/k$ and the right at $0.7q^2/k$. The prescribed dens.ty was 0.3 particles per site. From uniform initial conditions the system was allowed to relax to a steady state, after which the temperature profile was computed by averaging over about 1600 stations along the wall and 2048 time steps. Computations were made for three Knudsen numbers, 0.44, 0.22, and 0.10; the profiles, for rule set 1, are shown in Figure 12. As in the case for real gases, the profiles are linear with temperature jumps at the walls. Further, as expected, the magnitude of the jumps decline with Kn, approaching zero in the continuum limit. Figure 3.13 shows the temperature profiles for the first two cases computed with rule set 2. The results for the two rule sets are seen to be similar.

It should be noted that in the free molecule flow limit, $Kn \to \infty$, the model does not work. In the absence of collisions, the zero speed particles populations cannot adjust properly to those emitted from the walls to yield the correct mean temperature. This appears to be an inherent flaw in the model at this condition.

The time to reach steady state increases roughly as the square of the wall spacing, a result which suggests a similarly parameter of the form $y/\sqrt{\nu t}$ where y is the distance normal to the walls and ν an appropriate diffusion coefficient. In summary, similarities between the automaton results and those in real gases include:

1. linearity of the steady state profile
2. increasing temperature jump at the wall with increasing Knudsen number
3. existence of a similarity parameter of the form $y/\sqrt{\nu t}$

3.3.4. The Normal Shock Wave. Another classic test of molecular models is the normal shock wave. The particle population ratios in a box (Figure 3.14) are set to correspond to a certain temperature, density, and momentum in one direction, the momentum in the other direction being zero. The particles are allowed to relax to equilibrium by making the box doubly periodic. After that, at time zero, the boundaries perpendicular to the direction of mass motion are made specularly reflective (the other walls may be either periodic or specular). Subsequently, a shock forms at one end and propagates away from the wall at a speed that depends on the initial Mach number. At the other, a rarefaction wave forms.

By varying the initial temperature and velocity, shocks of different strengths can be generated and their characteristics computed.

Temperature and density profiles for several Mach numbers computed with the two rule sets are given in Figures 3.15 through 3.18. Here, in the stronger shocks, where the flow is further from equilibrium, results for the two rule sets may be expected to differ considerably. Again, as in the preceding example, the qualitative behavior is similar to that in real gases. In particular, (1) the density ratio increases and the thickness declines with initial Mach number, and (2) the temperature rise precedes the density rise.

Quantitative comparisons with the behavior of an ideal gas with a specific heat ratio of two, the proper value for a two-dimensional gas, have not yet been done because of several uncertainties. First the thermodynamic properties of the lattice gas, *i.e.*,, the specific heats, are difficult to work out. Next the Galilean-non-invariance of the model precludes a comparison of the jump conditions in the present unsteady flow to those of the steady shock. Finally, there is a degree of arbitrariness in the definition of the mean free path. In the case of molecules having a continuum of velocities, all collisions resulting in a deflection of more than a certain threshold are counted in calculating the mean free path, but in the present model, a particle is deflected, if at all, by either 45 degrees or 90 degrees. So, while counting only the non-zero deflection collisions gives a larger than correct mean free path, counting all collisions *i.e.*, even those collisions which leave the particle velocities unchanged, errs in the other direction.

3.3.5. The Arrow of Time. It is interesting to note that since each operation on the automaton is reversible, so are the macroscopic processes it models. More precisely, this is true when there is no external forcing, as in the shock and expansions wave flows discussed above*. Therefore, if at any time during the unsteady wave computation all molecular velocities are reversed, the system returns to its initial state. How is this behavior in accord with the second law of thermodynamics? This section describes a brief investigation of this question.

The one-dimensional flow computed in §3.4 was repeated for a shorter box and longer lengths of time, so that with reasonable computer time the approach to a stationary mean state could be examined. Figure 3.19, between times $t = 0$ and $t = 600$, shows the decay of the mean velocity in the box as a function of time. Between the same times Figure 3.20 shows the increase in entropy of the system as a whole as the gas comes to rest. If, as already noted, at any time, say $t = 600$, all molecular velocities

* In the heat conduction example, since the temperature at the walls is imposed at each instant of time, the procedure of reversal is less clear.

are reversed, the system reverses and exactly retraces the forward path as indicated in Figures 3.19 and 3.20 between $t = 600$ and $t = 1200$.

To see, nevertheless that the system has a preferred direction in time, we consider the effect of a small "error" on the system over-all mean velocity when it moves forward and backward in time. First, at $t = 600$, molecules at 0.1 percent of the sites in the box are reversed. The subsequent history is virtually the same as the original, and there are only microscopic differences. However, when the same error is then introduced at $t = 600$ as the system runs backward in time, *i.e.*, 99.9% of the particles are reversed instead of 100%, the effects are drastically different, as is illustrated in Figures 3.21 and 3.22. There is no resemblance to the macroscopic history of the system running backward with no error.

This behavior, and presumably that of real systems also microscopically reversible, may be summarized by saying that the solution describing the state history is *stable* in the forward direction but highly unstable in the reverse. A different way of looking at this behavior is that the CA evolution is an iterated Boolean map possessing a macroscopic equilibrium state. From yet another point of view, the automaton is a nonlinear system whose mean state is extremely sensitive to initial conditions in one direction in time but not in the other.

This short and incomplete study suggests that the cellular automaton could be a new tool for examining the irreversible behavior of reversible systems and the further resolution of the Loschmidt and Zermelo paradoxes. There is, of course, an enormous literature on these subjects but it has not yet been examined for relationships to the present work. A recent review by Coveney[13] describes many current ideas.

3.4. Conclusion - CA.

This work should be considered to be an exploration of the usefulness of cellular automata in the study of gas dynamics. It is premature to discuss the accuracy of the model (as the identification of several important difficulties makes clear) or of relative computation speeds. The primary usefulness of the model may, in fact, turn out to be as a mechanism for investigating concepts.

4. Appendix I - The Equilibrium State

The equilibrium equations have been solved in a few simple cases. In particular, for $u = v = 0$,

$$n_0 = n \left(1 - \frac{kT}{q^2} \right)^2$$

$$n_2 = n_4 = n_6 = n_8 = \frac{n}{4}\,\frac{kT^2}{q^2} \tag{A1}$$

$$n_1 = n_3 = n_5 = n_7 = \frac{n}{2}\,\frac{kT}{q^2}\left[1 - \frac{kT}{q^2}\right] .$$

The solutions for small velocity in one of the directions x or y are obtained by a linearization about the above stationary states. For small u, and $v = 0$, let $e_t \equiv kT + u^2/2$, $kT = e$,

$$n_0 = n\left[1 - \frac{e_t}{q^2}\right]^2$$

$$n_1 = \frac{n}{2}\left[1 - \frac{e_t}{q^2}\right]\left[\frac{e_t}{q^2} + \frac{u}{q}\right]$$

$$n_2 = n_8 = \frac{n}{4}\,\frac{e_t}{q^2}\left[\frac{e_t}{q^2} + \frac{u}{q}\right]$$

$$n_3 = n_7 = \frac{n}{2}\,\frac{e_t}{q^2}\left[1 - \frac{e_t}{q^2}\right] \tag{A2}$$

$$n_4 = n_6 = \frac{n}{4}\,\frac{e_t}{q^2}\left[\frac{e_t}{q^2} - \frac{u}{q}\right]$$

$$n_5 = \frac{n}{2}\left[1 - \frac{e_t}{q^2}\right]\left[\frac{e_t}{q^2} - \frac{u}{q}\right] .$$

By symmetry, the distribution for small v, $u \equiv 0$ is similar.

For the case $u = v$ but small, and $e_t = kT + \frac{u^2}{2} + \frac{v^2}{2}$, $kT = e_t$

$$n_0 = n\left[1 - \frac{e_t}{q^2}\right]^2$$

$$n_1 = n_3 = \frac{n}{2}\left[1 - \frac{e_t}{q^2}\right]\left[\frac{e_t}{q^2} + \frac{u}{q}\right]$$

$$n_2 = \frac{n}{4}\left[\frac{e_t}{q^2} + \frac{u}{q}\right]^2$$

$$n_5 = n_7 = \frac{n}{2}\left[1 - \frac{e_t}{q^2}\right]\left[\frac{e_t}{q^2} - \frac{u}{q}\right] \qquad \text{(A3)}$$

$$n_4 = n_8 = \frac{n}{4}\left[\left[\frac{e_t}{q^2}\right]^2 - \left[\frac{u}{q}\right]^2\right]$$

$$n_6 = \frac{n}{4}\left[\frac{e_t}{q^2} - \frac{u}{q}\right]^2 .$$

Experimental simulations tend to show the behavior as expected from the above analysis. Shown in Figure 4.7 is the stagnant particle density in a box of gas at $kT = 0.50$ and $u = v = 0$. Note the equilibration with time. The initial state was obtained by a random placement of particles in amounts giving the above values of kT, u and v.

5. Appendix II - Sound Propagation

The equations describing propagation of weak disturbances can be found by linearizing the Boltzmann equations for the model. The result is a set of first order linear partial differential equations which can be combined into a single equation for any variable, ϕ (a population density, for instance). The equation is

$$\left[\frac{\partial^2}{\partial t^2}\left[\frac{\partial^2}{\partial t^2} - q^2\frac{\partial^2}{\partial x^2}\right]^2 + \frac{\partial}{\partial t}\left[f_{11}\frac{\partial^4}{\partial t^4} - q^2 f_{12}\frac{\partial^4}{\partial t^2 \partial x^2} + q^4 f_{13}\frac{\partial^4}{\partial x^4}\right]\right.$$

$$\left. + \left[f_{21}\frac{\partial^4}{\partial t^4} - q^2 f_{22}\frac{\partial^4}{\partial t^2 \partial x^2} + q^4 f_{23}\frac{\partial^4}{\partial x^4}\right] + \frac{\partial}{\partial t}\left[f_{31}\frac{\partial^2}{\partial t^2} - q^2 f_{32}\frac{\partial^2}{\partial x^2}\right]\right]\phi = 0 \ , \qquad \text{(A4)}$$

where

$$f_{11} = 2\left[1 + \frac{kT}{q^2} - \left[\frac{kT}{q^2}\right]^2\right] \qquad f_{12} = \left[2 + 4\frac{kT}{q^2} - 3\left[\frac{kT}{q^2}\right]^2\right]$$

$$f_{13} = \frac{kT}{q^2}\left[2 - \frac{kT}{q^2}\right] \qquad f_{21} = \left[1 + 4\frac{kT}{q^2} - 4\left(\frac{kT}{q^2}\right)^2\right]$$

$$f_{22} = \left[1 + 2\frac{kT}{q^2} + \left(\frac{kT}{q^2}\right)^2 - 3\left(\frac{kT}{q^2}\right)^3\right] \qquad f_{23} = \left(\frac{kT}{q^2}\right)^3\left[1 - \frac{kT}{q^2}\right] \tag{A5}$$

$$f_{31} = 2\frac{kT}{q^2}\left[1 - \frac{kT}{q^2}\right] \qquad f_{32} = \frac{kT}{q^2}\left[1 - \left(\frac{kT}{q^2}\right)^2\right]$$

Equation (A4) represents a hierarchy of waves. Those of the higher order are important at early times and the disturbances propagate along characteristics of the highest order term, while at long times the lowest order waves prevail. For a discussion of equations of this type see Whitham[14]. The speed of propagation of the disturbances at long times is seen to be $q\sqrt{(1+kT/q^2)/2}$.

6. Bibliography

1. Gatignol, R., "Théorie cinétique des gaz à répartition discrète de vitesses," *Lecture Notes in Physics* Vol. 36, Springer-Verlag, Berlin, 1975 .

2. Frisch, U., Hasslacher, B. and Pomeau, Y., "Lattice-gas automata for the Navier-Stokes equation", Phys. Rev. Letters Vol. 56, 1986, 1505.

3. Bird, G. A., *Molecular Gasdynamics*, Clarendon Press, Oxford, 1976.

4. Carleman, T., *Problèmes mathématiques dans la théorie cinétique des gaz*, Uppsala, 1957.

5. Broadwell, J. E. 1964a Study of rarefied shear flow by the discrete velocity method. Jour. Fluid Mech. Vol. 19, 1964, 401.

6. Broadwell, J. E. 1964b Shock structure in a simple discrete velocity gas. Phys. Fluids Vol. 7, 1964, 1243.

7. Meiburg, E., Comparison of the molecular dynamics method and the direct simulation Monte Carlo technique for flows around simple geometries, Phys. Fluids Vol. 29, 1986, 3107.

8. Bird, G. A., Direct simulation of high-vorticity gas flows, Phys. Fluids Vol. 30, 1987, 364.

9. Broadwell, J. E., "Study of rarefied shear flows by the discrete velocity method," Report 9813-6001-RU000, Space Technology Laboratories, Inc., 1963.

10. Hardy, J., de Pazzis, O. and Pomeau, Y., Phys. Rev. A Vol. 13, 1976, 1949.

11. Frisch, U., D'Humières, D., Hasslacher, B., Lallemand, P., Pomeau, Y., and Rivet, J. -P., Lattice gas hydrodynamics in two and three dimensions, Complex Systems, Vol. 1, 1987, 649.

12. d'Humieres, D., Lallemand, P. and Frisch, U., Lattice gas models for 3d hydrodynamics, Europhys. Lett., Vol. 2, 1986, 291.

13. Coveney, P. V., The second law of thermodynamics: entropy, irreversibility and dynamics, Nature, Vol. 333, 1988, 409.

14. Whitham, G. B., Some comments on wave propagation and shock wave structure with application to magnetohydrodynamics, Comm. Pure and Appl. Math. Vol. XII, 1959, 113.

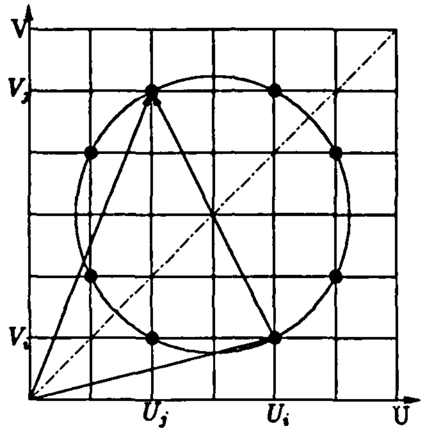

Figure 2.1. Reflections about 0/1, 1/1 and 1/0 are possible. Parity: EE, Relative velocity: (-2,4).

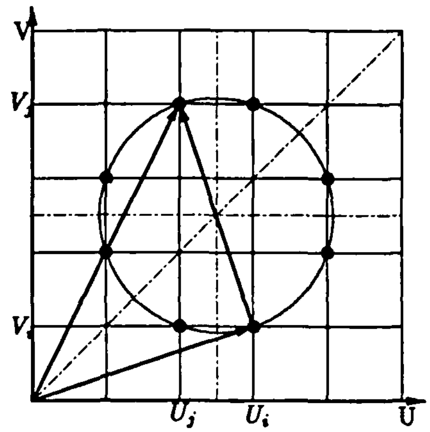

Figure 2.2. Reflections about 0/1, 1/1 and 1/0 are possible. Parity: OO, Relative velocity: (-1,3).

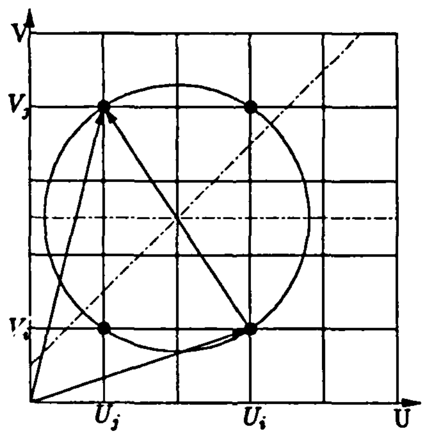

Figure 2.3. Reflections about 0/1 and 1/0 only are possible. Parity: EO, Relative velocity: (-2,3).

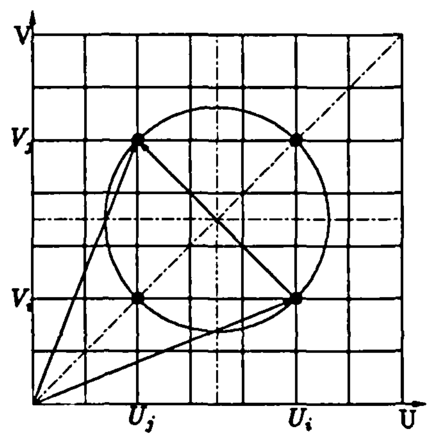

Figure 2.4. Reflections about 0/1 and 1/0 are possible; reflection about 1/1 is degenerate. Parity: OO, Relative velocity: (-3,3).

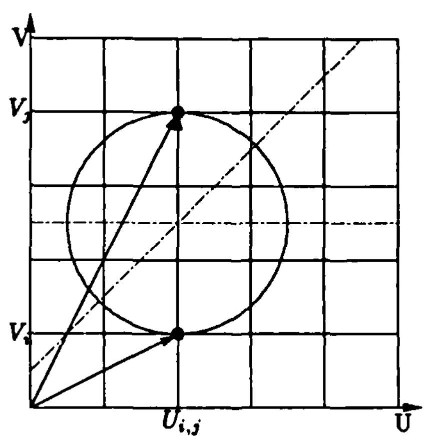

Figure 2.5. Reflections about 0/1 and 1/0 are degenerate. Parity: EO, Relative velocity: (0,3).

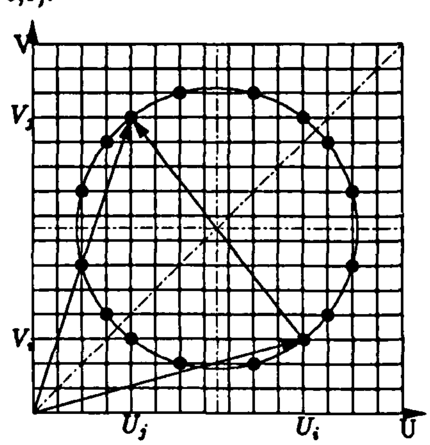

Figure 2.6. Reflections about 0/1, 1/1, 1/0, 2/1 and 1/2 are possible. Parity: OO, Relative velocity: (-7,9).

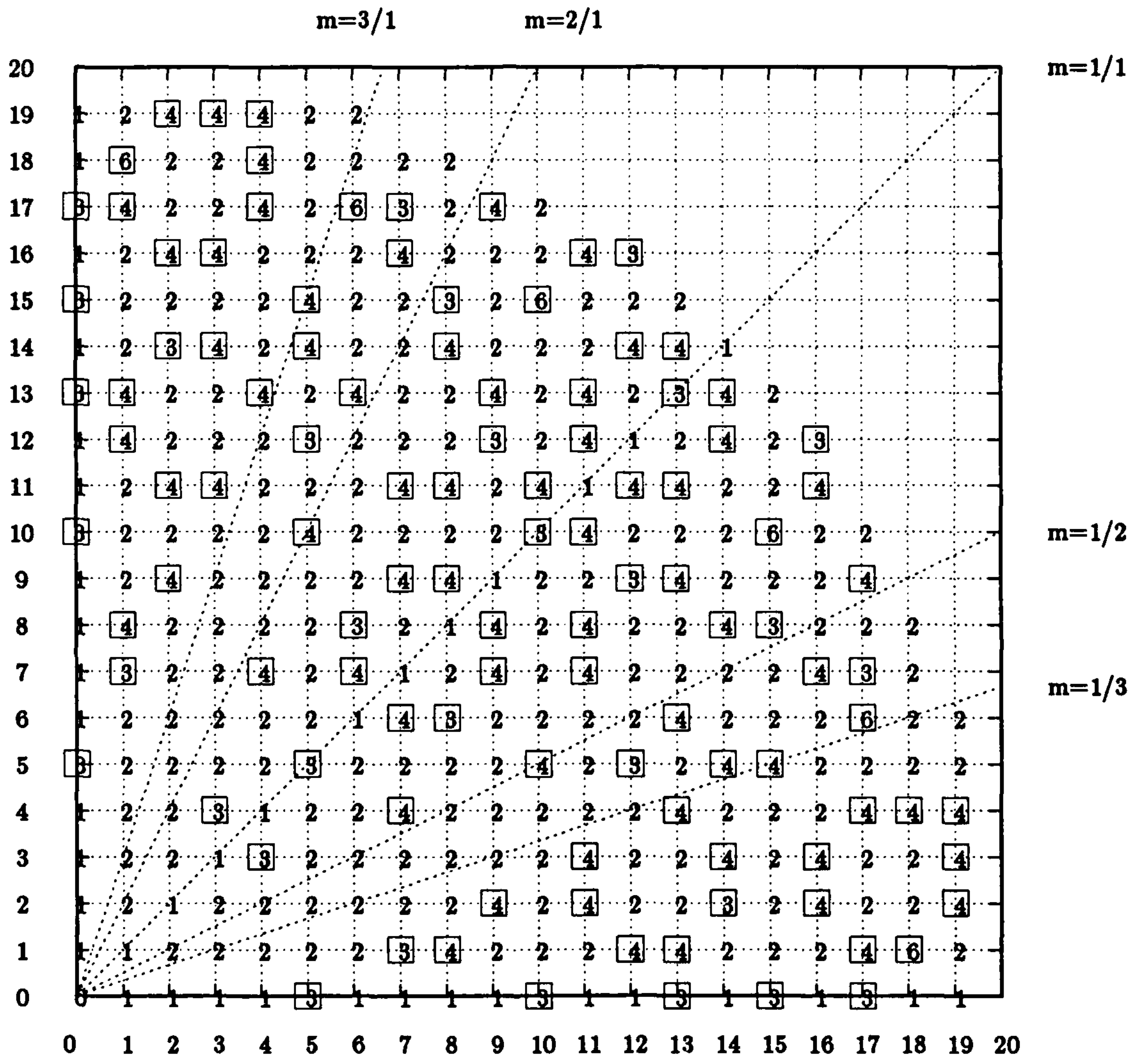

Figure 2.7. Each number indicates the number of points in the quarter of the plane which lie on a circle centered at (0,0) and which passes through the given point. Lines of slope 1/3, 1/2, 1/1, 2/1 and 3/1 are also shown.

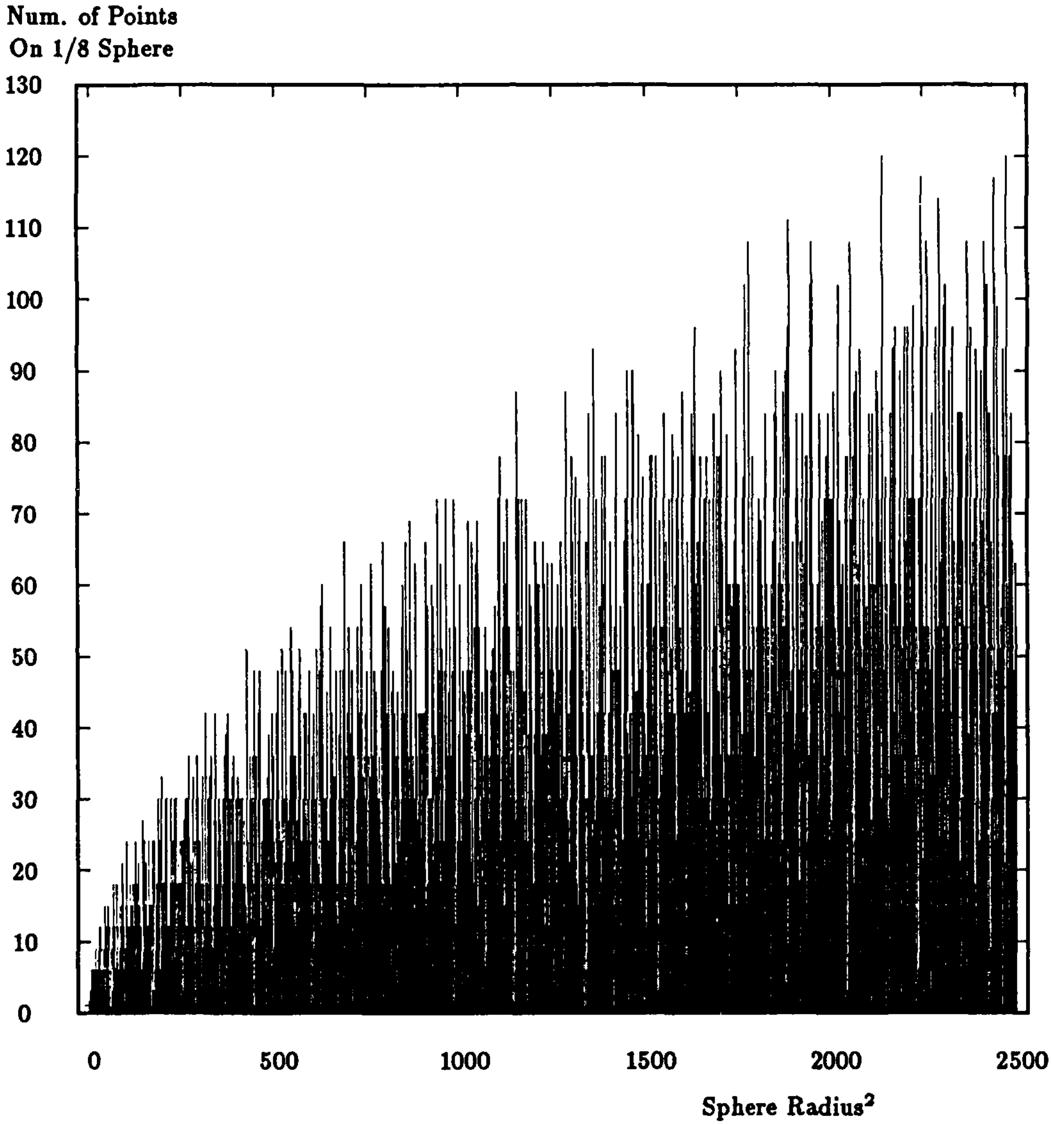

Figure 2.8. The number of the discrete points in the first octant on all spheres whose radius is less than 50.

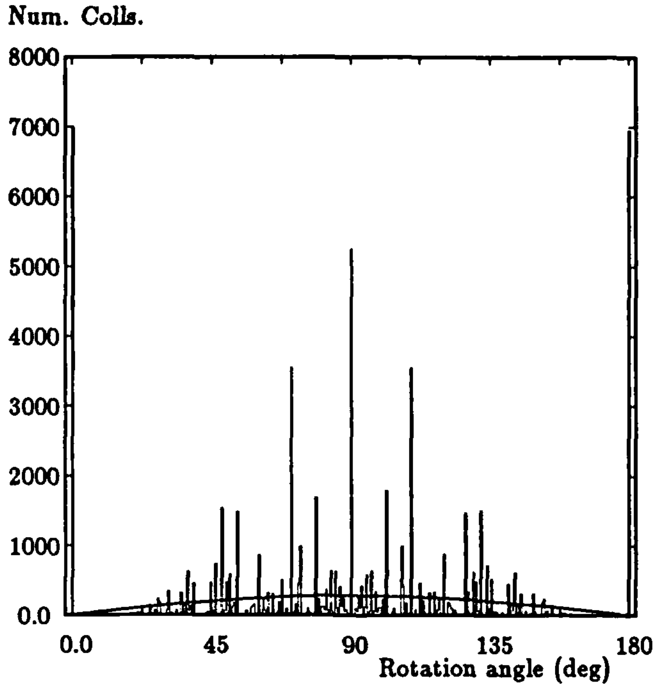

Figure 2.9a. Histogram distribution function of the rotation angle of the relative velocity vector in each collision. The solid line is the theoretical distribution. The particle temperature was RT $= \frac{\pi}{2}q^2$.

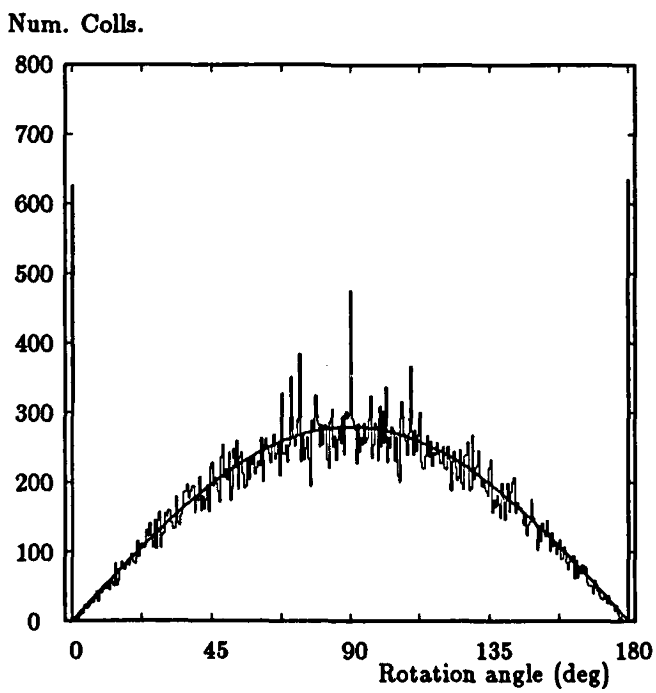

Figure 2.9b. Histogram distribution function where the particle temperature was RT $= 201.3q^2$.

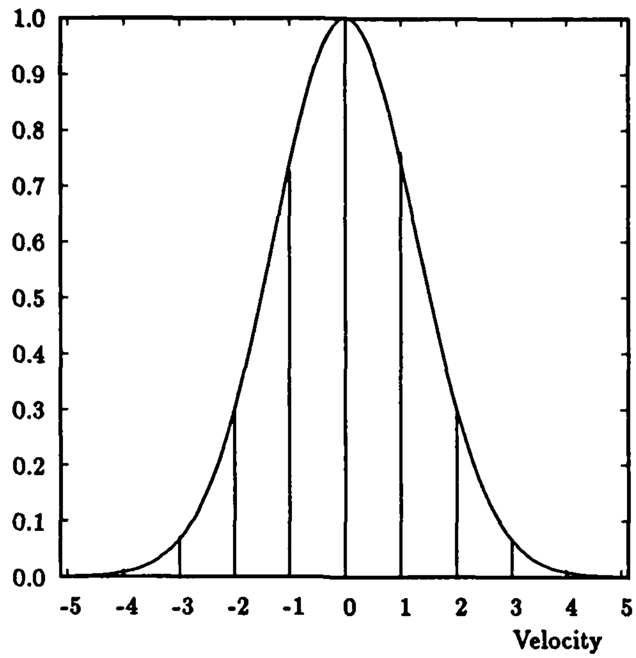

Figure 2.10a. Velocity component distribution function. The solid line is the theoretical distribution normalized to cover the same area as the discrete distribution. The particle temperature was RT $= \frac{\pi}{2}q^2$.

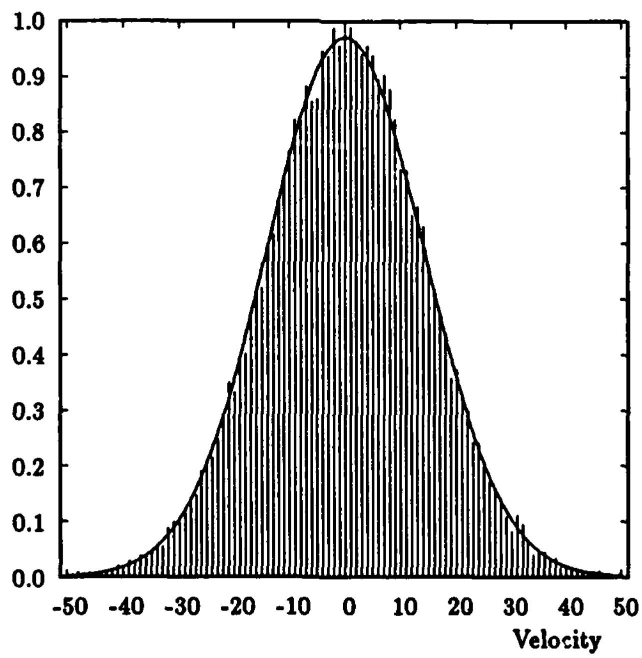

Figure 2.10b. Velocity component distribution function for RT $= 201.3q^2$.

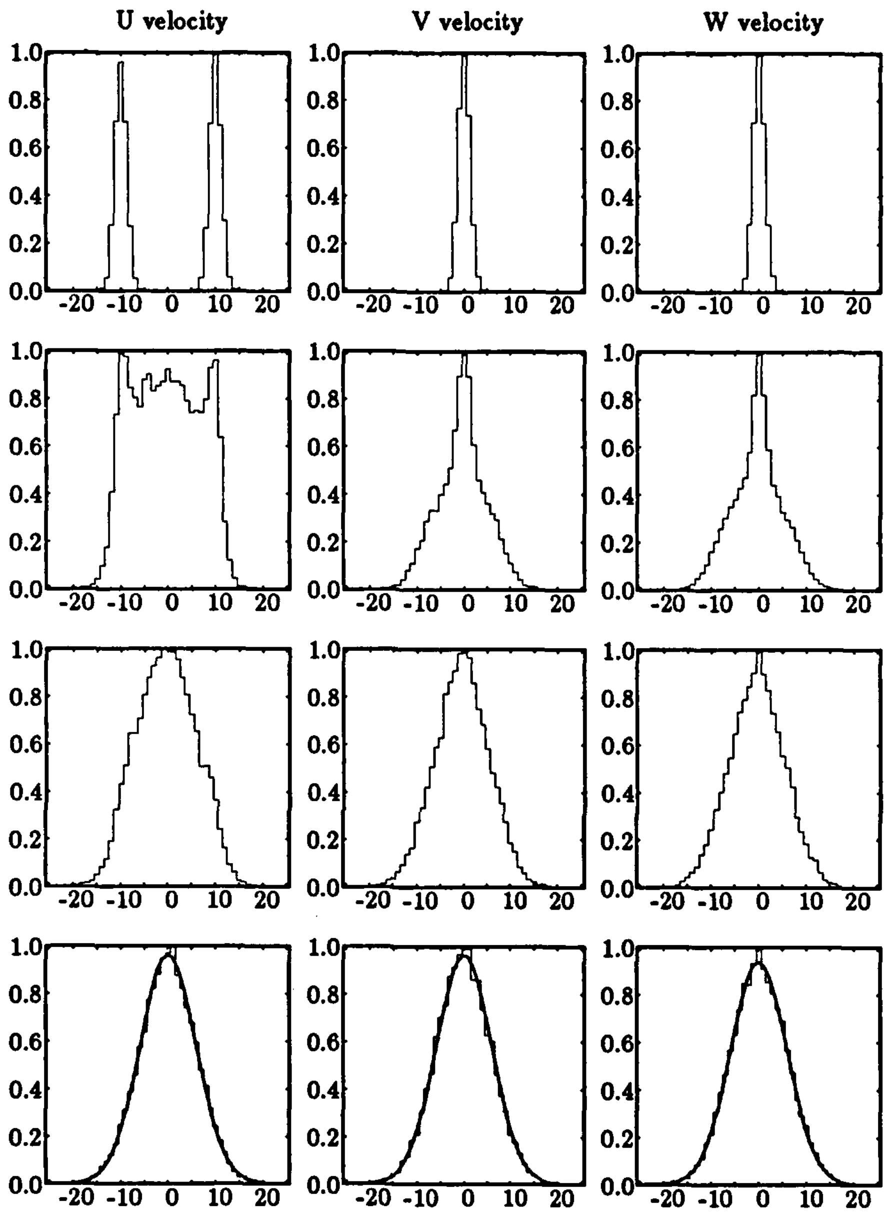

Figure 2.11. Conventional DSMC method. Time development of velocity distribution functions. The initial bimodal distribution spreads to form a Maxwellian. Equilibrium Maxwellians are also drawn in the last row for RT = $35.02q^2$. The first row is before any collisions, the second after 1 collision/particle, the third after 2 collisions/particle and the last after 10 collisions/particle.

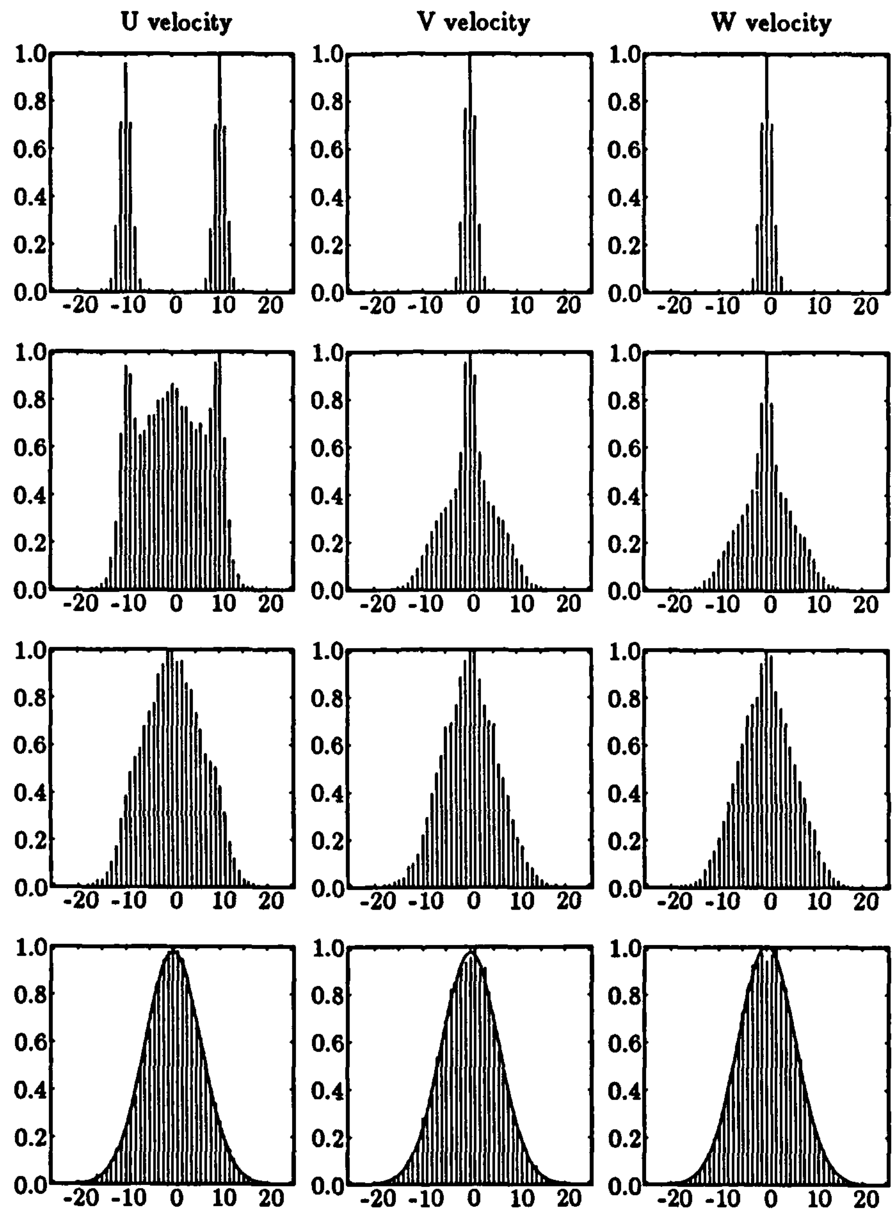

Figure 2.12. IDSMC method. Time development of velocity distribution functions. The initial bimodal distribution spreads to form a Maxwellian. Equilibrium Maxwellians are also drawn in the last row for RT = $35.02q^2$. The first row is before any collisions, the second after 1 collision/particle, the third after 2 collisions/particle and the last after 10 collisions/particle.

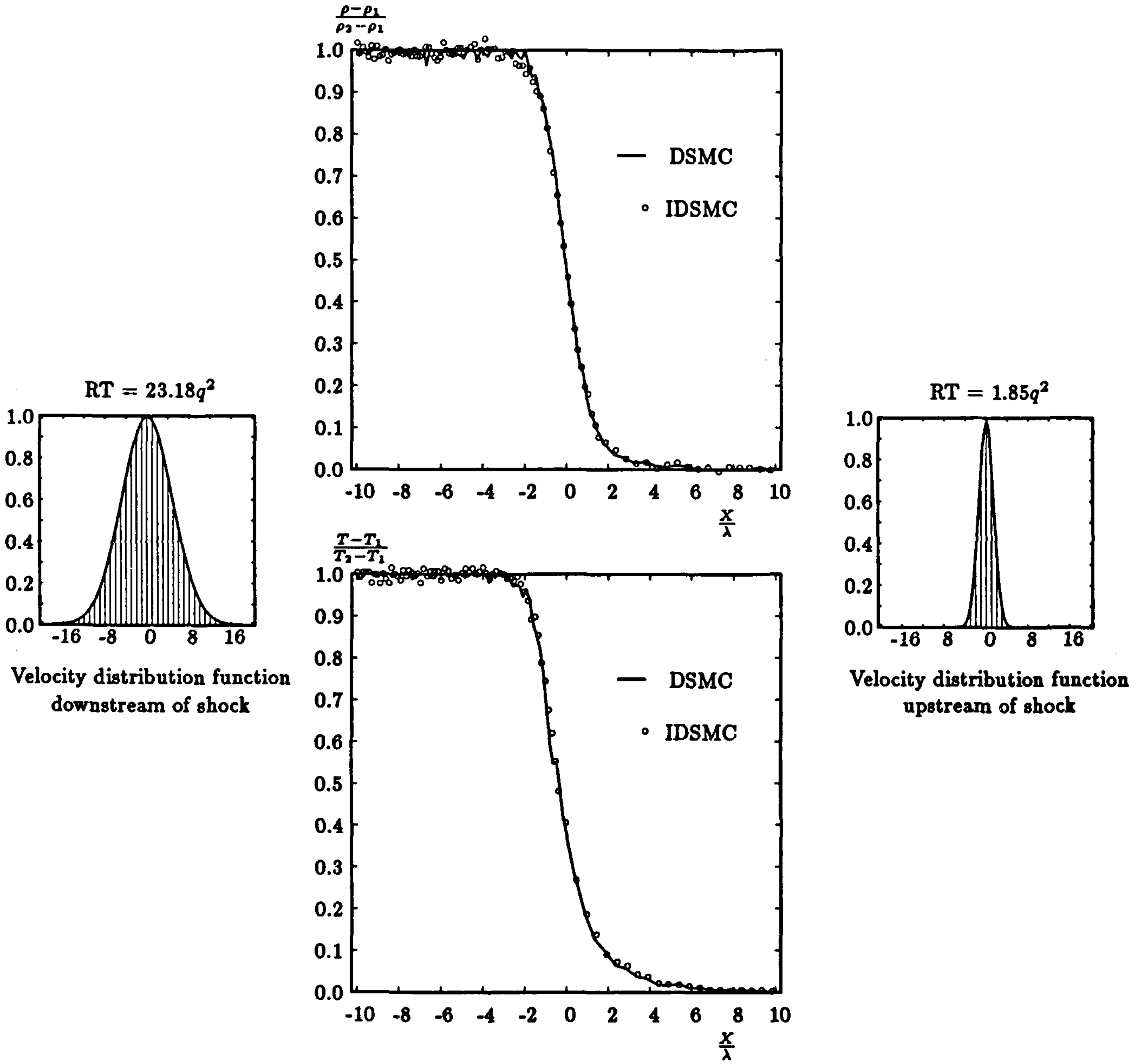

Figure 2.13. $M_s = 6.11$ Shock wave density and temperature profiles.

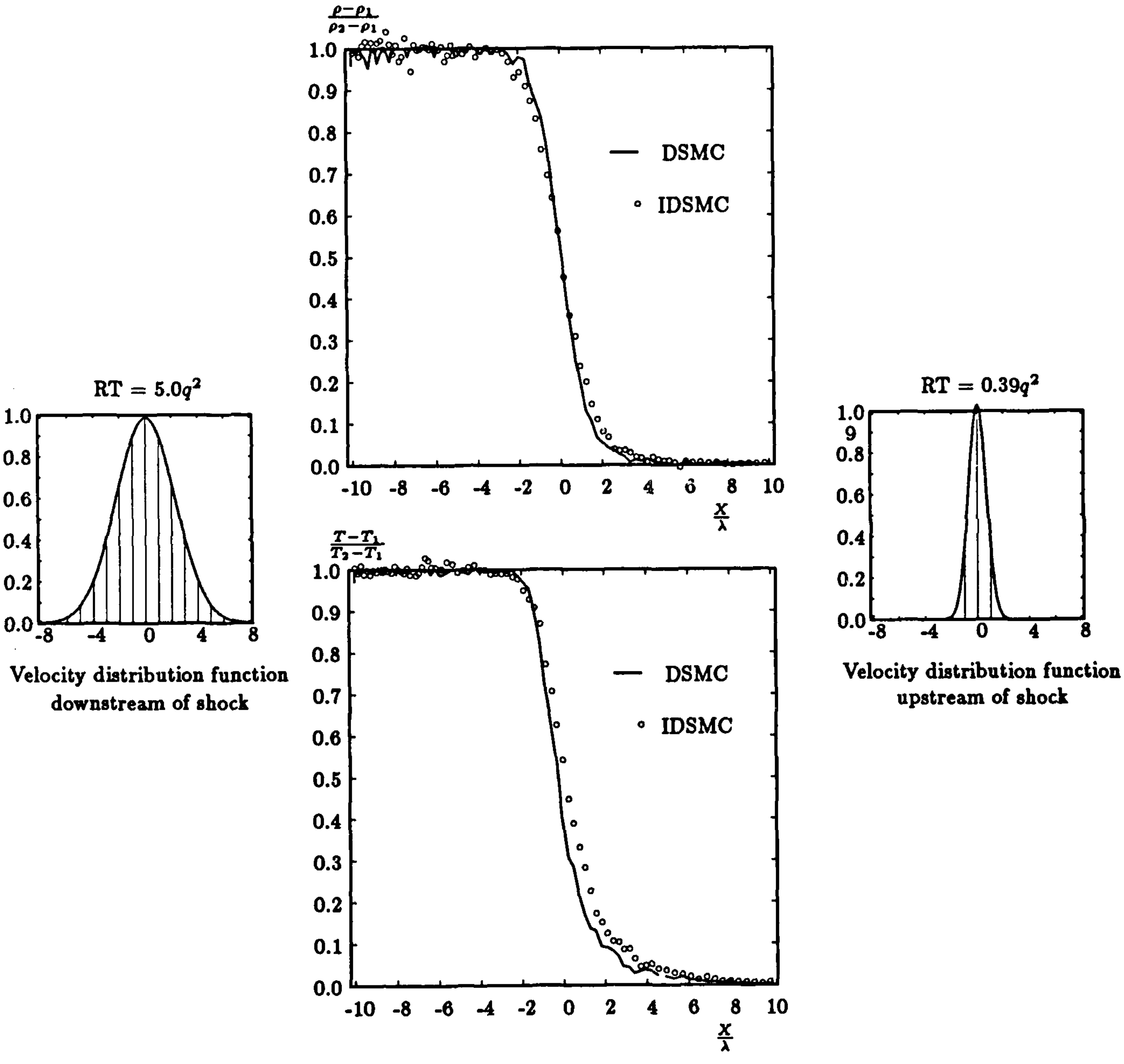

Figure 2.14. $M_s = 6.16$ Shock wave density and temperature profiles.

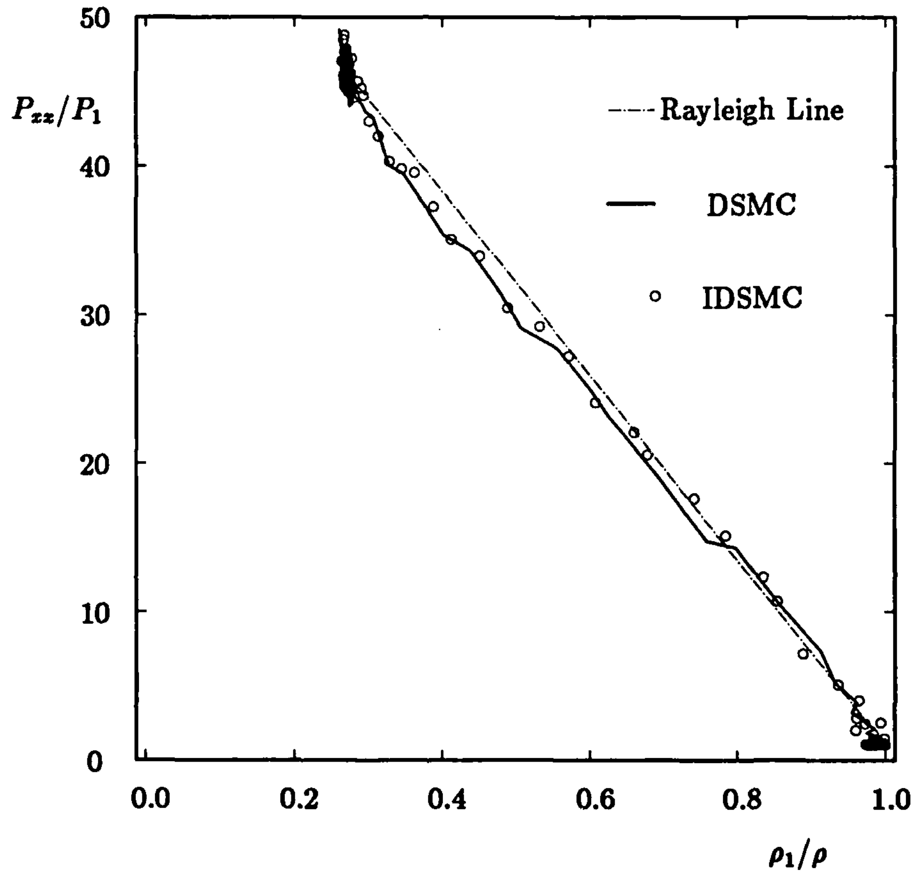

Figure 2.15. $M_s = 6.11$ Pxx variation through shock.

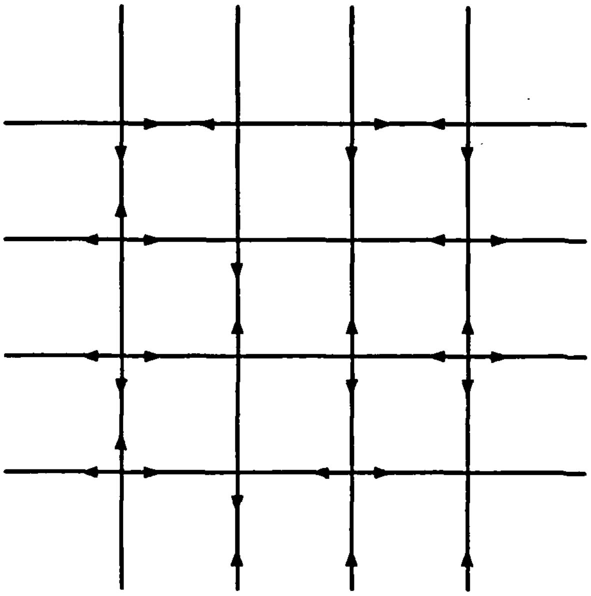

Figure 3.1 The lattice for the four velocity HPP model. All lattice links are of equal length and all particles have the same speed. A particle traverses one link length in one unit of time.

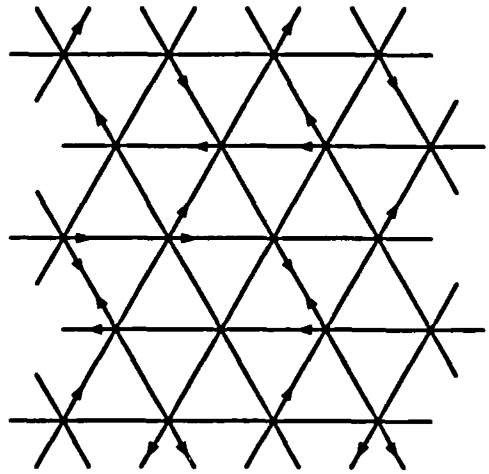

Figure 3.2 The FHP model: All particles have the same speed. Better symmetry compared to HPP, but spurious conservations with binary collisions alone.

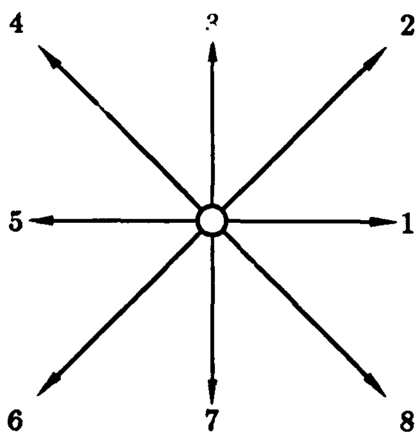

Figure 3.3 The nine velocity model. Four particle types have unit speed, four other particle types have $\sqrt{2}$ units of speed and one particle type has zero velocity.

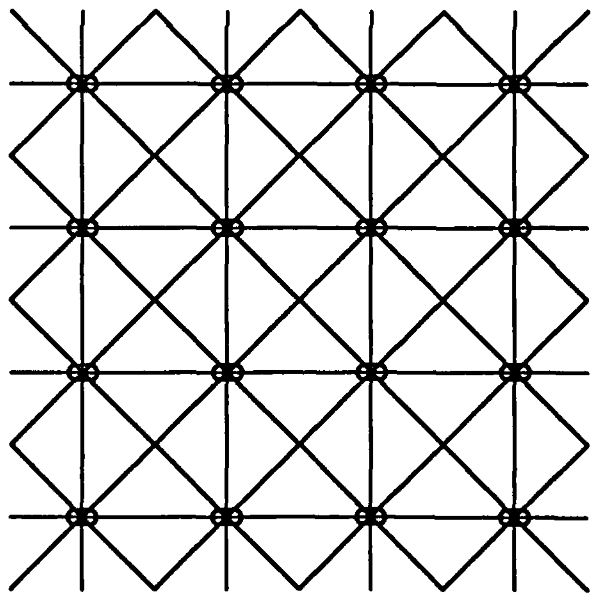

Figure 3.4 The lattice for the nine velocity model: Slow moving particles move along the horizontal and vertical links while the fast moving particles move along the diagonals.

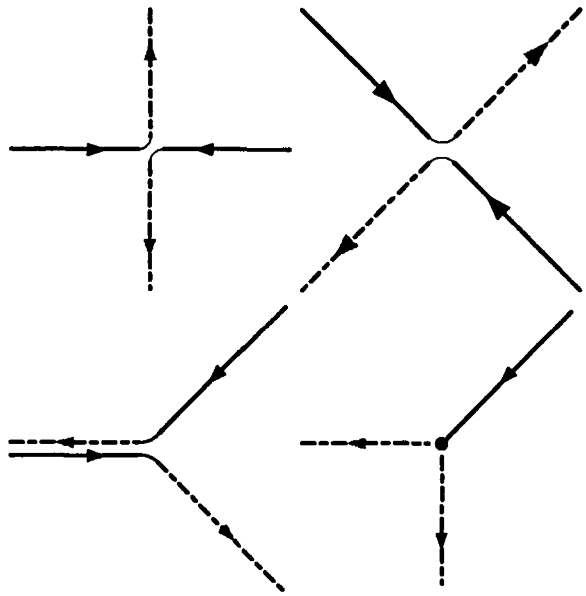

Figure 3.5 The binary collisions in the nine velocity model.

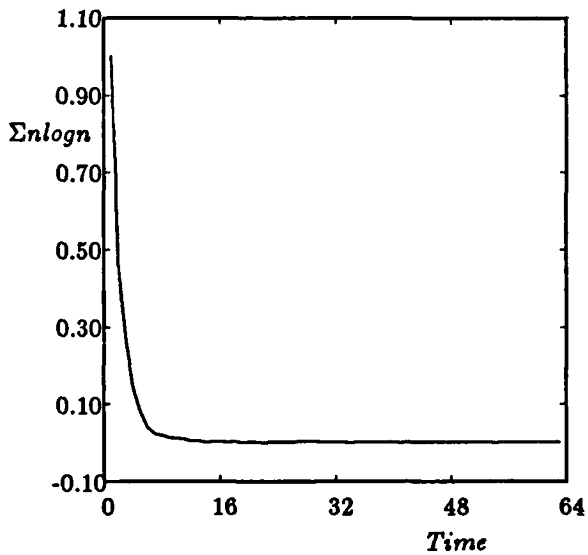

Figure 3.6 Evolution of $H = \Sigma n \log n$ with time for a box of lattice gas. Hydrodynamic quantities are constant with time and the initial state was chosen randomly.

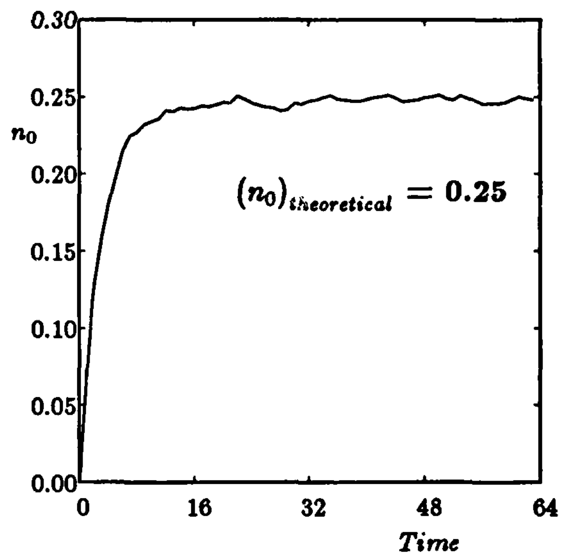

Figure 3.7 Evolution of n_0, the stagnant particle ratio with time for a box of the lattice gas ($kT = 0.5q^2$; $u = v = 0$; $\rho = 0.3$) shows that there is a unique equilibrium distribution of the particles.

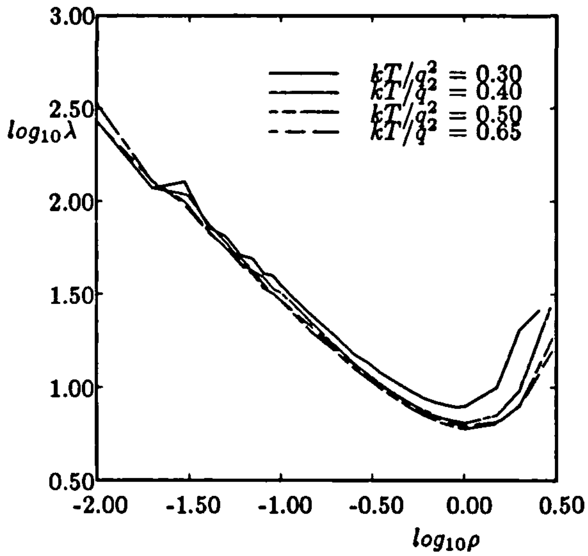

Figure 3.8log-log plot of λ vs ρ using Rule set 1. Note the unphysical increase of mean free path with density above a certain density and the variation of λ with temperature.

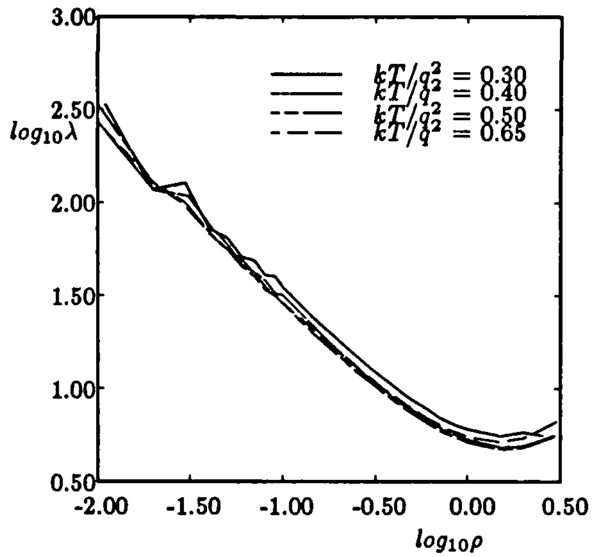

Figure 3.9log-log plot of λ vs ρ using Rule set 2. Note the similarity of the curves with those in **Figure 3.8** at low densities and the marked deviations at higher densities.

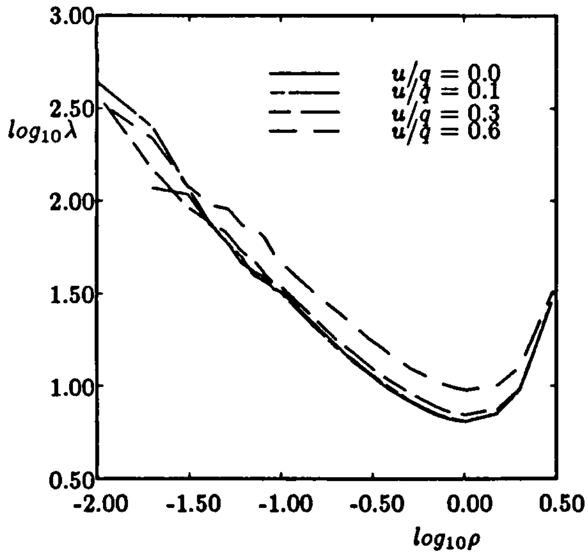

Figure 3.10log-log plot of λ vs ρ at different velocities using Rule set 1. $kT/q^2 = 0.40$ in all the cases. An illustration of Galilean non-invariance. At velocities upto 0.25, the variation in λ due to velocity seems to be small.

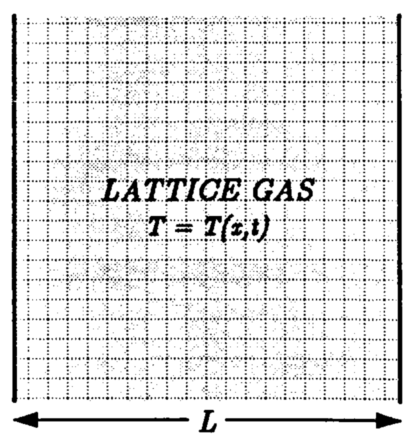

Figure 3.11 THE HEAT CONDUCTION PROBLEM: The two walls are maintained at different temperatures and the temperature profile in the lattice gas is sought. The left wall is maintained at $kT/q^2 = 0.60$ while the right wall is maintained at $kT/q^2 = 0.70$. Also the effect of the interwall spacing L is studied.

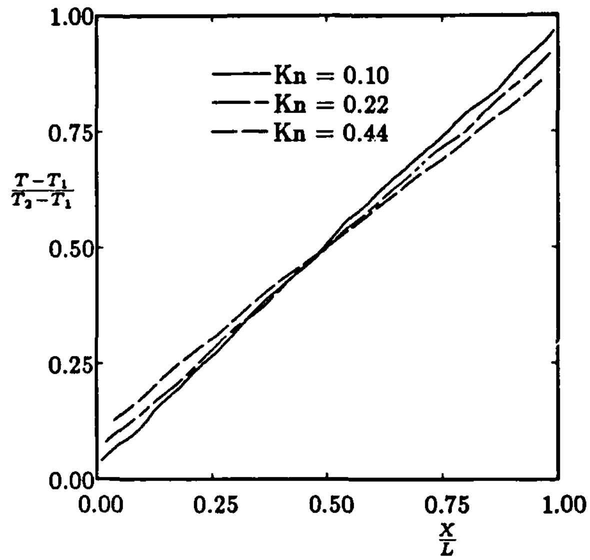

Figure 3.12 The steady state temperature profiles at three different Knudsen numbers. The average density is 0.3 particles in all the three cases. Rule set 1 is used.

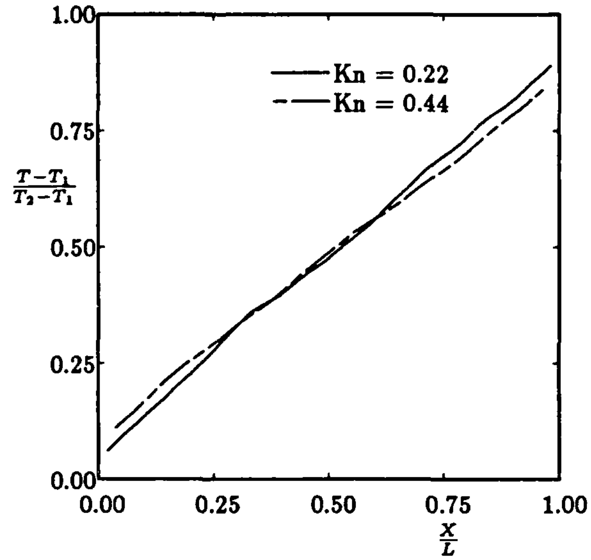

Figure 3.13 The steady state temperature profiles using Rule set 2. Conditions are the same as before. Note similarity to **Figure 3.12**

SPECULAR WALLS

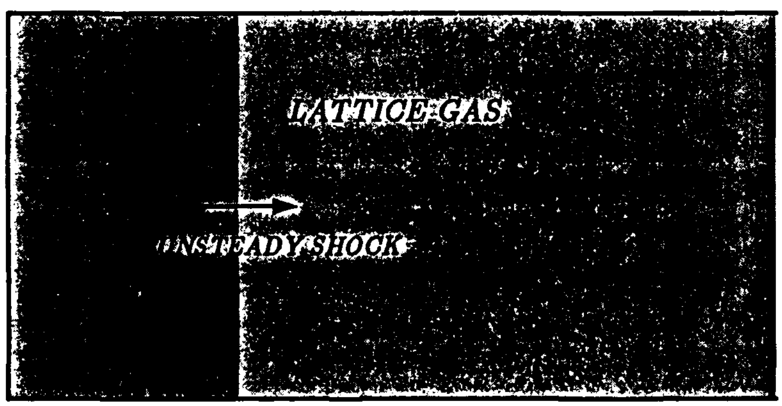

Figure 3.14 SHOCK PHENOMENA: Flow of the lattice gas into the closed end of a tube produces a shock. Shocks of different strengths can be formed by having different initial states.

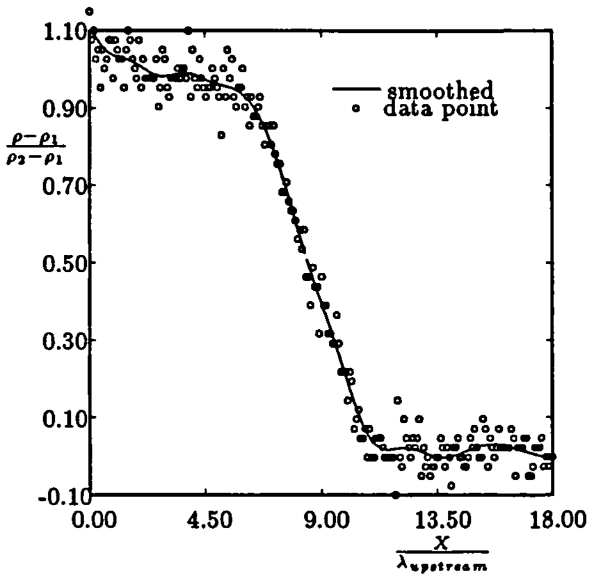

Figure 3.15 The density profile across a 1.13 Mach shock at an average density of 0.3 particles per site. Rule set 1 was used. The density ratio $\frac{\rho_2}{\rho_1} = 1.12$. The smoothing was done by using a low pass filter.

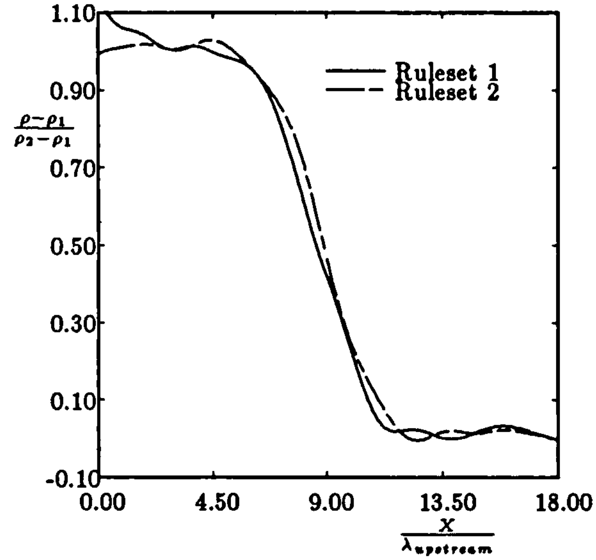

Figure 3.16 The (smoothed) density profiles across a 1.13 Mach shock at an average density of 0.3 particles per site using Rule set 1 & Rule set 2. The density ratio $\frac{\rho_2}{\rho_1} = 1.12$.

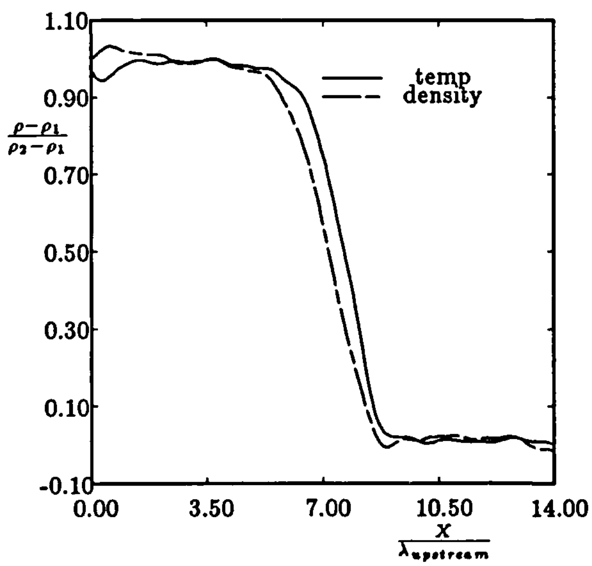

Figure 3.17 The (smoothed) density and temperature profiles across a 1.48 Mach shock. Note that the temperature shock propagates ahead of the density shock. The density ratio $\frac{\rho_2}{\rho_1} = 1.47$.

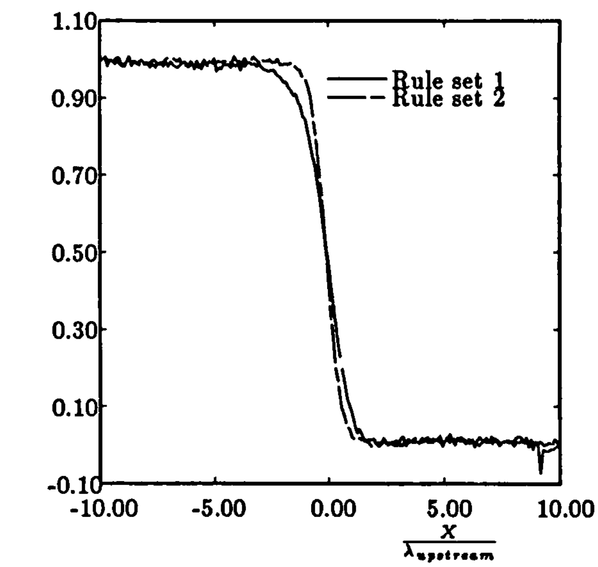

Figure 3.18 Density profiles across a 1.73 Mach shock using the two Rule sets at an average density of 1 particle a site. At these high densities, the two sets of rules simulate different viscosities. The density ratio $\frac{\rho_2}{\rho_1} = 1.71$.

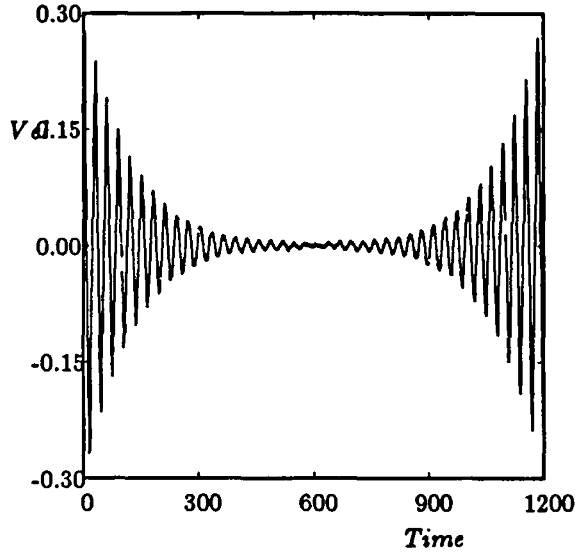

Figure 3.19 The mean velocity in a box decays with time. On full reversal at t=600, the system retraces its path and the initial state is fully recovered.

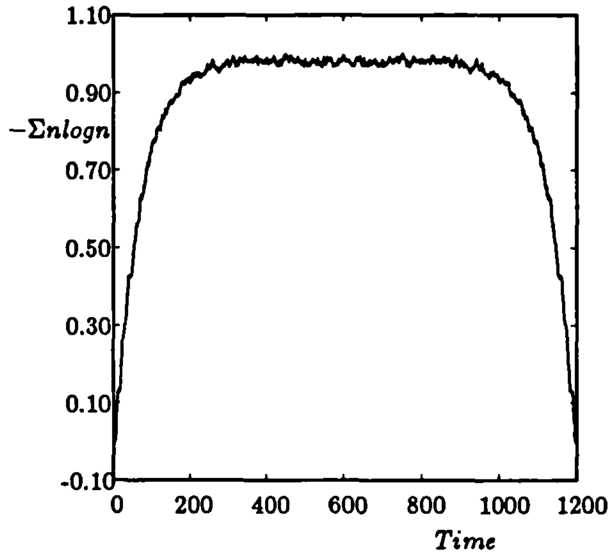

Figure 3.20 Entropy increases as the system equilibrates between t=0 and t=600, but decreases going backwards between t=600 and t=1200.

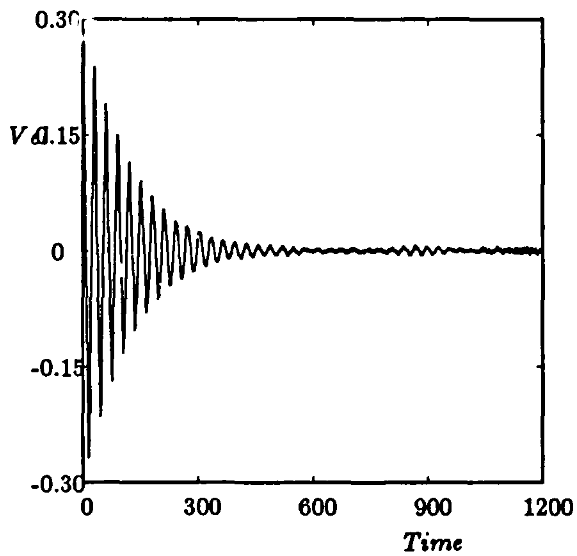

Figure 3.21 A small error in the reversal at t = 600 effectively prevents any recovery of the macroscopic velocity unlike in Figure 3.19

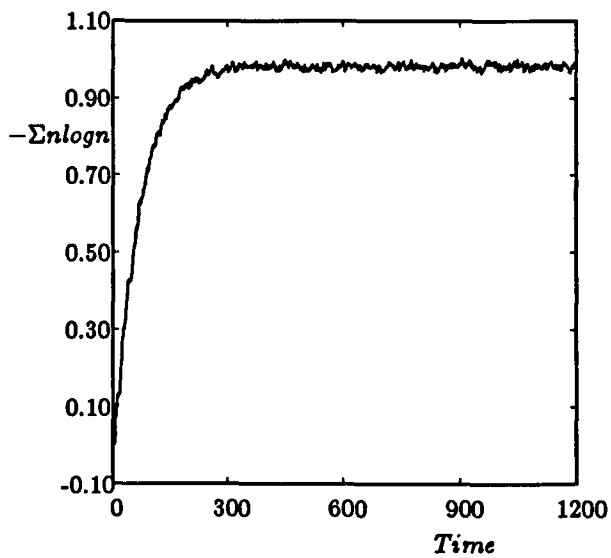

Figure 3.22 The small error at t=600 shows the instability of the reversed path. The entropy no more decreases like in Figure 3.20